November 1980
USA $2.95/DM 9.80

kilobaud

MICROCOMPUTING T.M.

To Stop a Thief!

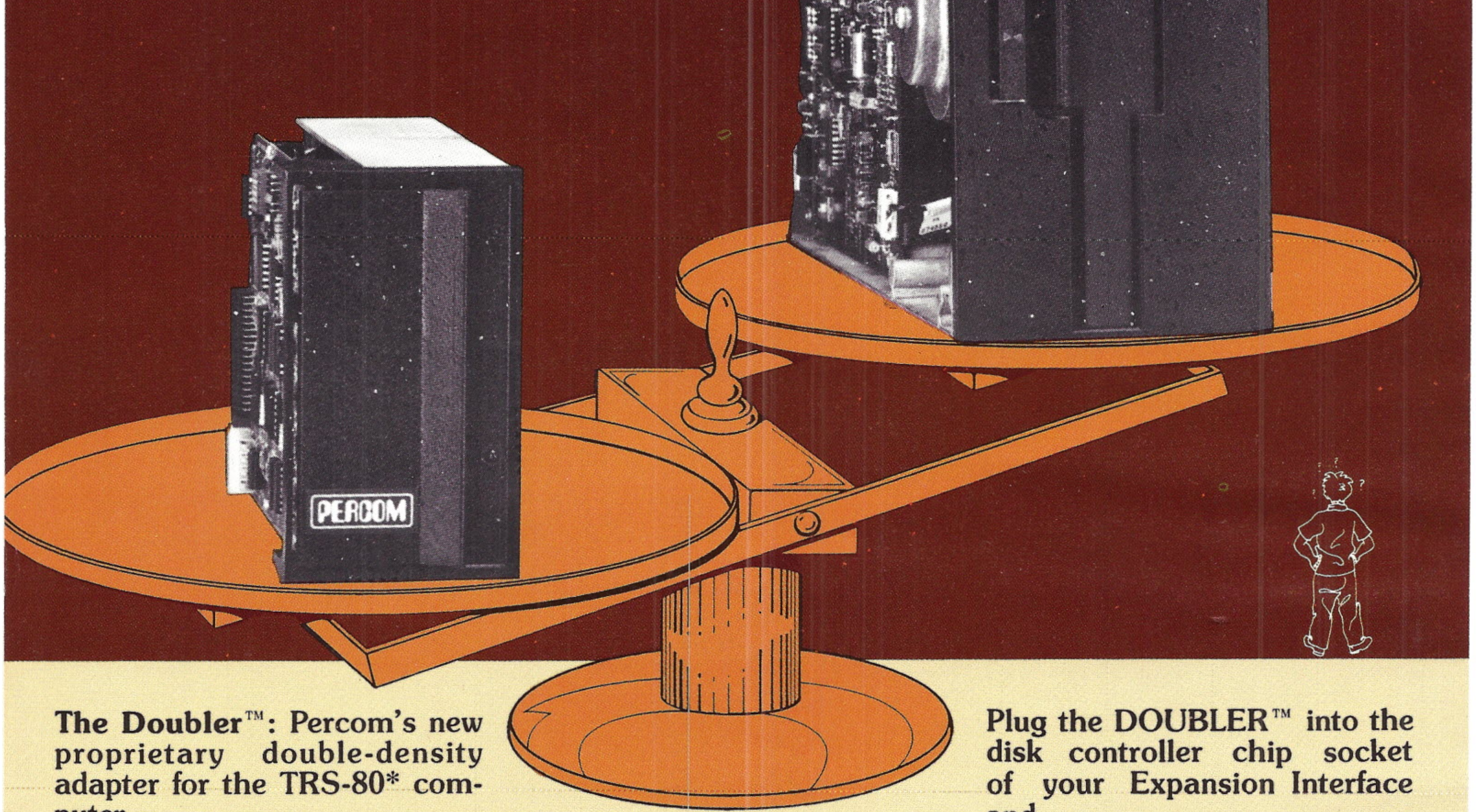
Safeguarding Your Computer Files

□Home Security and Status System□Micros as Business Tools□Video Graphics Primer□H8 Printer Interface□Z-80 Mailing Label Program□Tinkering with Tiny Basic□MC6809 System Design□PET Switch Control□Apple Printer Graphics □More on The Source□Exploring the ASCII Code. . . and more.

11
0 74470 65945

TRS-80* Model I Computer Owners

Store More Data on a 5″-Disk Than on an 8″-Disk



The Doubler™: Percom's new proprietary double-density adapter for the TRS-80* computer.

Plug the DOUBLER™ into the disk controller chip socket of your Expansion Interface and . . .

Store up to 354 Kbytes of formatted data on five-inch disks.†

- Increase **formatted** storage capacity of your minidiskettes from 1½ to almost 4 times.
- Use with standard 5-inch drives rated for double-density operation.
- The DOUBLER™ reads, writes and formats **either** single- or double-density disks.
- Proprietary design allows you to continue to run TRSDOS*, NEWDOS‡, Percom OS-80™ or other single-density software **without making any changes** to software or hardware.

Mini-Disk Systems

More storage capacity, higher reliability — from Percom, the industry leader. One-, two- and three-drive configurations in either 40- or 77-track format, starting at only $399.

- Includes DBLDOS,™ a **TRSDOS* compatible** double-density disk operating system.
- CONVERT utility, on DBLDOS™ minidiskette, converts files and programs from single- to double-density or double- to single-density.
- **Plug-in installation:** No strapping. No trace cutting. Restore your Expansion Interface disk controller to original configuration by simply removing the DOUBLER™ and reinstalling the original disk controller chip.

PERCOM DISCOUNT COUPON
worth $20
toward
The Purchase of a
DOUBLER™
Coupon No. 80M103
Expires December 30, 1980
Void where prohibited by law.

- The DOUBLER™ circuit card includes high-performance data separator, write precompensation circuits for reliable disk read operations — even on 77-track drives.

Introductory price, including DBLDOS™ and format conversion utility on minidiskette, **only** $219.95. Use the coupon for even greater savings.

Call toll-free, 1-800-527-1592, for the address of your nearest dealer, or to order direct from Percom.

†Percom TFD-200™ drive, OS-80D™ operating system

PERCOM

PERCOM DATA COMPANY, INC.
211 N. KIRBY GARLAND, TEXAS 75042
(214) 272-3421

PRICES AND SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE.

™ trademark of Percom Data Company, Inc.
* trademark of Tandy Radio Shack Corporation which has no relationship to Percom Data Company.
‡ trademark of Apparat Company, Inc.

SUPER BRAIN QD™

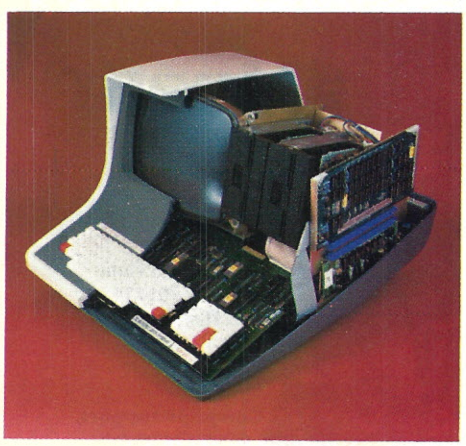
Once in a great while someone comes along with a simple improvement for an already great product. Take our SuperBrain, for example. Really a simple concept. A high-powered, low cost microcomputer packaged in an attractive desk top cabinet. So how do you improve on that?

WE DID IT...

It wasn't enough that our SuperBrain had such standard features as twin double density 5¼" drives with over 300,000 bytes of disk storage. A full 32K of dynamic RAM - expandable to 64K in seconds. A CP/M* Disk Operating System which assures compatibility to literally hundreds of application packages presently available. A crisp, 12" non-glare screen with a full 24 line by 80 column display. A full ASCII keyboard with a separate keypad and individual cursor control keys. Twin RS232C serial ports for fast and easy connection to a modem and/or a printer. And, dual Z80 processors which operate at 4 MHZ to insure lightning-fast program execution. No, it wasn't enough. So we made it better.

ANNOUNCING SUPERBRAIN QD...

Our new QD model has all of the features of our phenomenally popular SuperBrain with the addition of double-sided disk drives and an extra 32K of dynamic RAM. So, for only a modest increase in price, you can order your next SuperBrain with more than twice the disk and memory storage capability. But, best of all, the new QD model has the same tough, rugged construction and exceptional quality that made our SuperBrain such a success.



HOW DID WE DO IT?

The secret of SuperBrain QD's incredible disk storage lies within our new double-density double-sided disk drives. A total of nearly 720,000 bytes of data are formatted on two specially designed 5¼" drives. And that's more than enough to get you started with most serious small business applications. And SuperBrain QD's standard 64K of dynamic RAM will handle even the most complicated programming tasks.

Of course, if you're into megabytes instead of kilobytes, you may think neither SuperBrain is right for you. Not so! Intertec offers 20-96 megabytes of hard-disk storage which connects in seconds to either the SuperBrain or SuperBrain QD. So, your original investment is always protected. As you grow. No matter how much your needs expand.

BUT IS IT RELIABLE?

Our best salesmen are our present users. Not only have SuperBrain users been impressed with the inherent reliability of the system, they tell us that no other microcomputer system available today offers such a unique modular design concept. Just about the only tool required to easily maintain the system is a common screwdriver. And Intertec's total commitment to product service and customer support, with service outlets in most major cities, insures your original investment will be a valuable one for many years to come.

THE DECISION IS YOURS.

Whether your next SuperBrain is a regular model or our QD version, you will have the satisfaction of knowing you purchased what is becoming one of the world's most popular microcomputer systems. And regardless of which model you choose, you'll probably never outgrow it because you can keep expanding it.

So, call or write us today for more information. Intertec systems are distributed worldwide and may be available in your area now.

INTERTEC DATA SYSTEMS®

2300 Broad River Rd., Columbia, SC 29210
(803) 798-9100 TWX: 810-666-2115

196

*Registered trademark of Digital Research, Inc.

micro info

$ This symbol next to a title in the table of contents indicates that the article is a business-application article.

Manuscripts

Contributions in the form of manuscripts with drawings and/or photographs are welcome and will be considered for possible publication. We can assume no responsibility for loss or damage to any material. Please enclose a self-addressed, stamped envelope with each submission. Payment for the use of any unsolicited material will be made upon acceptance. All contributions should be directed to the *Microcomputing* editorial offices. "How to Write for Microcomputing" guidelines are available upon request.

Editorial Offices:

Pine Street
Peterborough NH 03458
Phone: 603-924-3873

Advertising Offices:

Elm Street
Peterborough NH 03458
Phone: 603-924-7138

Circulation Offices:

Elm Street
Peterborough NH 03458
Phone: 603-924-7296

To subscribe, renew or change an address:

Write to *Microcomputing*, Subscription Department, PO Box 997, Farmingdale NY 11737. For renewals and changes of address, include the address label from your most recent issue of *Microcomputing*. For gift subscriptions, include your name and address as well as those of gift recipients. Postmaster: Send form #3579 to *Microcomputing*, Subscription Services, PO Box 997, Farmingdale NY 11737.

Subscription problem or question:

Write to *Microcomputing*, Subscription Department, PO Box 997, Farmingdale NY 11737. Please include an address label.

Kilobaud *Microcomputing* (ISSN 0192-4575) is published monthly by Wayne Green, Inc., 80 Pine St., Peterborough NH 03458. Subscription rates in U.S. are $25 for one year and $53 for three years. In Canada: $27 for one year only, U.S. funds. Foreign subscriptions (surface mail)—$35 for one year only, U.S. funds. Foreign air mail subscriptions—$62 for one year only, U.S. funds. In Europe, contact: Monika Nedela, Markstr. 3, D-7778 Markdorf, W. Germany. South African Distributor: KB Microcomputing, PO Box 782815, Sandton, South Africa 2146. Australian Distributor: Electronic Concepts, Attention: Rudi Hoess, 55 Clarence Street, Sidney 2000, Australia. Second-class postage paid at Peterborough NH 03458 and at additional mailing offices. Phone: 603-924-3873.

kilobaud

MICROCOMPUTING T.M.

contents: November '80

ARTICLES

DEPARTMENTS

Cover photo by Paul Grupp; illustrations by Diana Shonk.

Wayne Green

PUBLISHER'S REMARKS

Business Microcomputers: Still a Rip-off?

One of the benefits—or crosses to bear, depending on how you look at it—of the job as editor of a computer magazine is to visit computer firms to see what they are doing. In recent months I have seen a lot of them. My question is: If microcomputers are so hot for small-business use, why don't I see them in use in our own industry? The message should be clear to any prudent person. It certainly makes me suspicious.

Perhaps the industry suffers from the shoemaker's kid syndrome, and there really are a lot of wonderful small-business systems available to do all those things promised. If this is so, I have another question begging for an answer. For several years I've been asking in my editorials for articles about successful business applications of microcomputers. I've also asked virtually every systems manufacturer to push their customers or dealers hard to get such articles written, pointing out that such articles would be solid gold in helping other dealers sell their products. Still, with all that pushing, you know how many articles we've had on business uses of micros? Very few.

Before I get into an explanation of what I perceive as the real situation, I would like to provide one more piece of evidence: an obvious lack of published business software by the thousand or so firms in the business. We see games, some educational programs (mostly painfully primitive), scientific programs (usually bordering on the ridiculously simple) and junk business programs such as financial calculations on loans and checkbook balancing.

As I have often written, Instant Software could make an instant rich man out of a programmer submitting a good business package for any specific industry. But what do we see? Precious little so far. Are other publishers doing much better? Not much. Most of the so-called business programs are embarrassing to the industry and have created more ill will than sales. I constantly hear about the incessant rip-offs due to lousy software being marketed.

There *are* some programmers writing good, usable software for micros, and there *are* some very happy customers, but I suspect that the quantity is pathetically small so far. A recent survey, made under questionable auspices, showed that only about 20 percent of the businesses are buying computers from computer stores.

With all due lack of respect, until I become more confident in the software available for business applications, I will certainly not go to a computer store and expect much help. If I, as a publisher of software, am unable to get programs of any significant value, how can individual stores with far fewer resources be expected to come up with good software?

In addition to evaluating the programs submitted to Instant Software, I also am in a position to evaluate the software being sold by many other firms. We receive them for testing from one of the magazines, or even go so far as to buy a copy, so we will be aware of the state of the art, which is in disarray. Many of the Radio Shack programs are disastrous and, I'm sure, are doing tremendous damage to the whole industry. But when you consider that a billion-dollar corporation is unable to do any better than some of the debris they have been selling, that, in itself, tells you something, if you are paying attention.

Most of the so-called business programs are embarrassing to the industry and have created more ill will than sales.

Of course, there is always the chance that I am so secluded in my ivory tower that I am in the midst of many fantastic business programs, but just don't know they exist. If that's true, I should expect some furious letters from livid programmers or dealers cursing me out. Well, there's a first for everything. However, after the cursing is over, I hope that a copy of a program they consider of value will be included so I can check it out. I will be more than delighted to eat my words.

A look at the historical development of microcomputers provides us with a simple explanation for why things are as screwed up as they are at present.

It was well over a year after the Mits Altair system was put on the market before there was even a usable language to go with it. And it took about the same length of time for the original hardware bugs to be worked out of the system for it to be of any serious use. By then we had the Imsai and several other S-100 bus systems on the market, most of which were in fair working order, but were without even BASIC at that time.

If we are going to have any significantly complex programs we have to have a very flexible and well-supported language, good hardware, including dependable disks, and an operating system to take care of most of the routine chores. The S-100 bus was developing well, with the CP/M operating system making it start to look really good. Then we had the first really serious blow to the whole field—the Heath H8, with its own bus and own BASIC. The PET made it worse, with a different BASIC and no operating system at all. Their endless delays in disk support staggered the field.

Next we had the TRS-80, with another BASIC and no operating system. These blows were sinking the S-100 systems and all of the support that had built up around them. There were music systems, talking systems, listening systems, control systems, plotters and modems for the S-100. The list was getting to be almost endless, and the star seemed bright for micros. All this went up in smoke when Heath, Commodore and Radio Shack hit the market with national advertising, distribution in thousands of stores and virtually bare-bones systems. This threw two or so years of hard work and product development for most of the industry right out the window.

Few of the major firms recognized what was happening. They tried to continue along with business as usual, ignoring the new entries and their incompatibility with the S-100 bus. These older firms could have made it if they had recognized what was happening. After visiting most of the defunct firms, I can tell you first hand what I saw—blindness. The heads of the firms, all brand-new millionaires as a result of the explosive growth of the industry, thought they knew more than anyone else and were not inclined to listen to outside advice. They were mostly surrounded by people anxious to curry favor by telling them that they were right. So down went The Digital Group and their arrogance. Down went Processor Tech and their lavish booths at shows. Down went Imsai and their lavish ads in *Byte*. Mits disappeared from sight after absorption by Pertec. And so it went.

It has not been long that the programmers interested in writing complex software for the field have had the tools with which to work. The TRS-80, being the most popular system, is being supported the best by new software, which is only reasonable. But how long have we had an extensive BASIC language and a really good operating system to use as a basis for developing business programs? Maybe a year at best. Well, it takes at least that long for someone to write a set of complex programs, check them out, get rid of most of the bugs, work up the documentation, put them into practice for a couple of months to find out how they work in actual use and then make the necessary changes.

The evaluation people at Instant Software tell me that they are beginning to see more and

more sophisticated program packages being submitted for publication and marketing. These are a lot tougher to check out than games and simple scientific or educational programs, but they are working hard on them. In some cases, they are setting them up with local businesses to see how they work in practice. I think we are starting to see good programs become available. This will mean that the promises made to small businessmen will someday soon become honest ones.

Word Processor Woes

Good business programs not only have to be able to do a lot of work for a businessman, but they also have to be easy to use and as self-prompting as possible.

A couple years ago, spurred on by the enthusiasm of a local computer store, I decided to check out a word processor. We paid over $8000 for an Algorithmics system. I never did get delivery of the entire system, and it took most of the two years to get it to work reliably. I can't even begin to tell you how frustrating it is to write a long article and then have the system unable to ever find it again on the disk. The support I got from the manufacturer was one of the more irritating aspects of the investment. They seemed to have an enormous death wish. Indeed, they have managed to barely eke out a living selling their product, while other firms were making millions . . . all by dint of being as resistant as possible to customer relations (I suspect). I figure that if they will do that to the editor of a major magazine, imagine what they will do to the average customer.

When the whole system was working—which *did* happen at times—it was still so cumbersome to use that I seldom used it. I doubt if I used it once a month on the average. Still, as I explained to them, my original intention was to buy the system to help out the dealer, test it for a few weeks and then sell it off after writing a nice article on it. The tests went from weeks to months as the system crashed and I waited for delivery of the monitor (it was never delivered, even though it was paid for in full). Finally, the process stretched out to years, with the maker (Seals) of the computer and disk system involved going out of business. There were memory problems, bugs in the program and serious printer problems. No businessman in his right mind would put up with that baloney for any time at all. And, if the man had any friends interested in getting something like that, he would do all in his power to prevent someone else from enduring similar suffering.

I finally gave up completely with the Algorithmics system and sent it over to our lab, where we have full-time technicians keeping the damned thing running. Some day I may be inveigled into trying another word processor, but the shock was so bad after that experience that it is going to take a while . . . and an awful lot of soothing salesmanship to iron out the emotion left over. Those repeated traumas live on.

The editing function of a word processor is convenient, but it is a luxury that I seldom need for my type of writing. I always edit the material I write for my magazines before I send it along to be set in type. But this is done with a pen and the usual editing marks. I would waste enormous amounts of time if I did all that editing on a word processor.

Another serious problem with a word processor is that you can see only a small amount of your material at any one time. If I put the rough editorial manuscript on my desk, I can flip from page to page quickly to make sure that I have not been too redundant in either style or content. This is much slower with a WP system. The copy on a tube is much slower to read, and that, as I'm sure you know, results in a concomitant loss of retention of the material. It is thus slow to read and slow to edit, with the results that I long ago gave up trying to use the WP for writing my editorials.

At first, when I had to write an important letter, I would type it on the WP, but after losing many such letters due to disk errors or due to problems with the printer, I found that was not an efficient system. Once the hardware and software are perfected, I may put another WP system on my desk and cautiously try it again, but it is going to take some time to wean me from my dependable IBM-60.

One of the problems that I had with the word processor was the complexity of the program, which made it necessary for me to always have a set of cards with notes on them next to the keyboard so I could remember the special coding system used.

The two big notebooks of documentation accompanying the word processor were almost impossible to figure out without an index. Whenever I got into some problem and the printer wouldn't work right or the whole thing would hang up, I would have to drag out the books and try to figure out what was wrong . . . and that could take an hour. The program had a system for putting page numbers on long letters, but I never could figure from the instructions how to get it to work.

The program had many functions which were of such little use that I would forget how to use them. It was possible to search for a specific word, but not once did I ever have the need for that. There was a function for moving blocks of text around, but either that was not working or I didn't know how to get it to work.

The chaps running the company were able to sit down at the system and make it do all sorts of fantastic things that I could never duplicate from the instruction books. Who wants to have to take a six-week course in how to use a new typewriter? Yet, without adequate instructions and constant use to refresh your memory, this is where the state of the art seems to be.

Until someone can produce a word processor as easy to use as a typewriter, I'm not going to be convinced of the value of word processing for the average office. I'm willing to try some more systems, but my experience with Algorithmics was a definite downer.

There is no way that I can personally test every word processor, so I'm asking for help from all readers. If any one of you has a word processor—either good or bad—you've been using, how about writing a frank evaluation of it? Either way, you'll help others and perhaps prevent people from getting the $8000 ream job that I feel I received. If others have used an Algorithmics and found it to be good, I'd even

kilobaud
MICROCOMPUTING T.M.

PUBLISHER/EDITOR
Wayne Green

ASSISTANT PUBLISHER/EDITOR
Jeff DeTray

ASSOCIATE PUBLISHER/DIR. PUBLICATIONS
Edward Ferman

MANAGING EDITOR
Dennis Brisson

ASST. MANAGING EDITOR
Susan Gross

COPY EDITOR
Eric Maloney

ADMINISTRATIVE ASSISTANTS
Cresca Clyne
Pat Graham
Nancy Noyd

ASSOCIATE EDITORS
Robert Baker
Ken Barbier
Frank Derfler, Jr.
Rod Hallen
Peter Stark
Sherm Wantz

MANUFACTURING MANAGER
Noel Self

PRODUCTION MANAGER/PUBLICATIONS
Nancy Salmon

ASST. PRODUCTION MANAGER
Michael Murphy

ART DIRECTOR
Diana Shonk

PRODUCTION DEPARTMENT
William Anderson, Jr.
Steve Baldwin
Pati Burr
Tedd Cluff
Linda Drew
Robert Drew
Bruce Hedin
Kenneth Jackson
Ross Kenyon
Maryann Metivier
Dion Owens
Robert Sawyer
Patrice Scribner
Susan Symonds
John White

PHOTOGRAPHY
William Heydolph
Terrie Anderson
Reese Fowler

TYPESETTING
Barbara Latti
Sara Bedell
Mary Kinzel
Linda Locke

EXECUTIVE VICE PRESIDENT
Sherry Smythe

CORPORATE CONTROLLER
Alan Thulander

EXECUTIVE ASSISTANT
Leatrice O'Neil

ACCOUNTING MANAGER
Knud Keller

CIRCULATION MANAGER
Debra Boudrieau

CIRCULATION
Barbara Block
Pauline Johnstone

BULK SALES MANAGER
Ginnie Boudrieau

PUBLIC RELATIONS
Joseph Wilson

ADVERTISING
603-924-7138
Kevin Rushalko, Mgr.
Marcia Stone
Hal Stephens

like to hear about that. The prospect of such a report seems remote.

The money I feel I wasted on the word processor is miniscule compared to the approximately $250,000 we have shoveled into the Prime computer—with hardly anything positive to show and net losses going into the millions of dollars directly attributable to inability to do what was promised.

But if a bunch of computer "experts" such as we have assembled can cause such horrible wastes of money, how can the average businessman be expected to get a good value from computers? This is what I hope to be able to accomplish with this and my other computer magazines. By getting mistreated consumers to write about their experiences, I hope to force the offenders to change and provide better equipment, services and programs. By publishing articles and letters about the good systems, we can put further pressure on the bad ones. This is up to *you*. If you find something which is good and do not write about it, you are helping the guys in the black hats. If you get screwed and sit by embarrassed in silence, you are as bad as the scoundrels who sucked you in.

Good or bad, let's hear from you.

New Ideas

The new Erwin International 10 megabyte Winchester technology disk drive has a built-in tape cartridge system for a four minute backup. It uses a seven-channel system and a standard, but little known, type of tape cartridge (3M DC100A). I wonder if there aren't some techniques which could be evolved to allow us to store those 10 megabytes on a regular cassette? We have four-channel tape heads available now at reasonable prices, and we could ship a C-60 through a fast forward in a couple of minutes. If that won't make it, perhaps we could use the helical recording head of our video recorders to get that data on there and off again.

Tape Formats

As I look over the articles published so far in *Kilobaud Microcomputing*, I notice a lack of articles on the subject of data storage on tape. Virtually all of the people who have had to work with tape recording of computer data tend to work empirically, rather than from a technical understanding of what they are doing. They try this, and then that, hoping for the best while waiting to see if the data loads. Even the "professionals" in the field are, for the most part, working by the seat of their pants.

I'd like to see some definitive articles on the cassette recording formats being used by today's major firms. These might explain why most of us had so much trouble with the early Radio Shack system, and how Personal Micro Computers can now sell a fast loader that works (most of the time) at the fast-forward speed, while Radio Shack's makes you wait ages for data to be loaded.

There is a need for information on the PET recording system, and an explanation of why

If you find something good and do not write about it, you are helping the guys in the black hats.

most PET computers have cassette recorders which are so far out of alignment that they are almost incompatible with tapes made to meet the PET test tape standard. More than 90 percent of the "unloadable" cassettes returned to Instant Software come from PET owners who are still unaware that their recorders are out of alignment with the PET standard. This late in the game, they are in a miserable position: They are unable to read their earlier tapes made on their own systems if they realign the head now. Yet, without this realignment, they are incompatible with the rest of the world.

Let's see some articles on what tones, what data formats and what recording techniques are being used today for all the systems in use.

Opportunity

Whether you realize it or not, with the microcomputing industry growing at a high rate, your career opportunities in this industry are excellent. You can prepare for this by learning all you possibly can about as many microcomputer systems as possible. You also want to learn all you can about programming, as well as about hardware. These skills can get you into the business. From there on it is up to you to learn all you can. You want to know about selling, about advertising, about managing people. The more you know and can do, the more unlimited your horizons.

As the industry grows, there are going to be more and more $50,000 jobs available, with the main problem being finding the people to fill them. You make money by going where the money is. Today this means microcomputers.

Getting a Job

The growth of our publications and, in particular, Instant Software has forced me to interview many people interested in working for us.

We're interviewing for editorial help, technical editors, programmers, technicians, carpenters, plumbers, middle management, typesetters, art productionists, data-processing people, salesmen and audio tape experts, so we have to talk with a lot of people. Frankly, I'm surprised at the number of people who obviously have given virtually no thought to what sort of an impression they are making.

If you are going to look for a new job, I have a few hints for you. First, there is the resume. I have seen virtually none of any value so far. Not one person in a hundred includes a photograph in the resume, so how is a personnel manager to remember one person from another once a dozen or two have been interviewed? A photo is very helpful.

Then comes the matter of creating a resume aimed at the firm and the job you want to get. This is not time for a general listing of your education and experience; you are *far* more than that. You have special skills that will be of value to the firm, and you should make sure that these are cited, complete with references to your education and experience to prove that you are indeed capable of doing what you say you can.

Remember the old saw: "You only have one chance to make a good first impression." This means writing a neat and concise letter to cover your resume. It also means that when you go for the interview you should look your very best. You'd be surprised at how many people don't even try to create a good first impression, and consequently lose out. Neatness definitely counts.

Sure, you are going to be nervous at an interview, but you want to come across the best you can. This means sticking to discussing things you know. If you try to exaggerate or lie, the chances are you'll muff it. It isn't difficult for the interviewer to see through baloney. Get as much information about the job or jobs the firms has open and see how you might be able to help them. If you come across as arrogant or unsure of yourself, you are not helping your cause.

Remember the Golden Rule: "Them with the gold make the rules." *You* are the one being interviewed, not the firm. *You* are looking for a good position with career possibilities. This is not the best time to play hard to get.

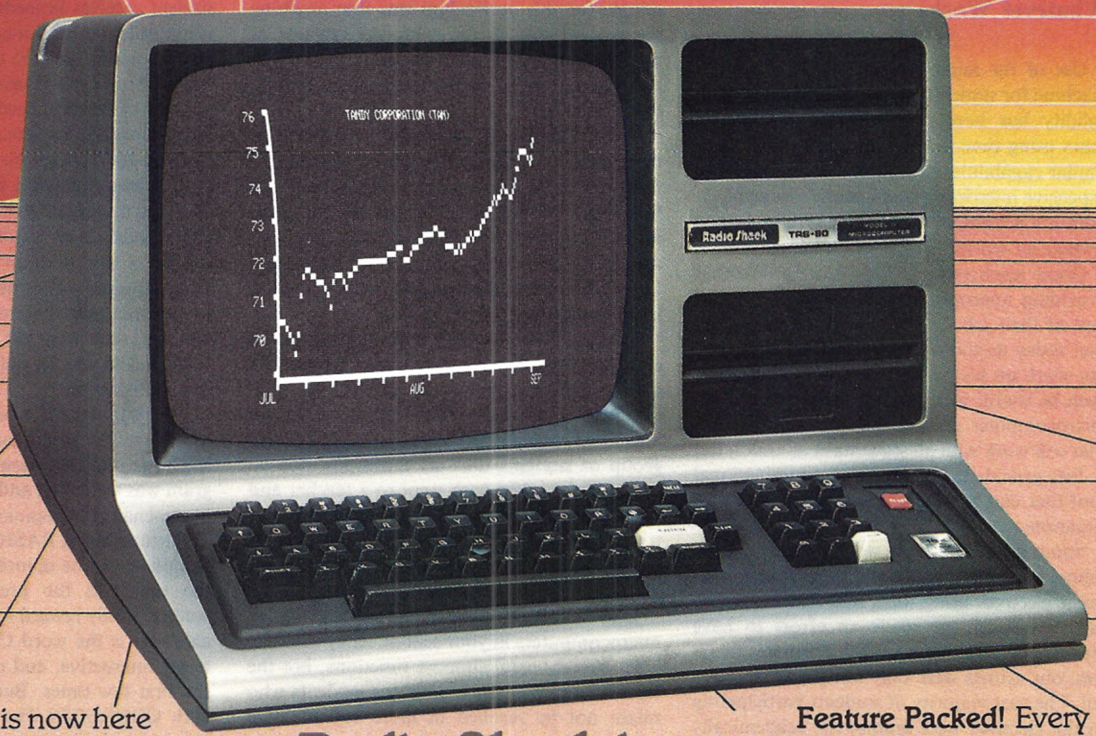
If the firm has a position that offers you a lot in terms of a career, remember that in turn *you* have a lot to offer the firm in achieving your career goals. The more successful you are, the more benefit you will be to the firm. In discussing these things during an interview, try to always put the emphasis on the benefits to the firm, not on what you want or need. I want Instant Software people who want to be with a successful company and help it to grow, not those who just want to move to New Hampshire to enjoy the mountains and fishing. I don't hire anyone because they need a job. I hire because I need some work done—and done well.

If you put all requests you make of the firm in terms of the benefits to the firm, you will go a long way toward getting what you want.

Winners, Winners!

If you attend a major microcomputer show, be sure to stop by the *Microcomputing* booth to say hello and to enter our free drawing. We're collecting names from every show we attend, and on July 4, 1981, we'll select one of them to win a Level II TRS-80. In addition, at every show, we award $100 worth of Instant Software to a lucky visitor. The winner from last August's Personal Computing '80 show in Philadelphia was Jon Wolfe of Clayton, NJ. Congratulations.

The TRS-80™ Model III. A New Standard in Personal Computers!



The new standard is now here — this beautiful, feature-packed, one-piece desktop computer system at a very, very affordable price. Continuing the TRS-80 tradition begun with our famous Model I, the amazing Model III gives you everything you've always wanted in a personal computer—including easy expandability.

It Talks Your Language! Model III is available with either Level I or powerful new Model III BASIC. Best news of all is that nearly all Model I software is compatible with Model III, so you already have a huge library of applications to choose from. Radio Shack already offers over 80 quality packages — from games to sophisticated business programs to word processing.

Big Storage Capacity! Model III BASIC features dual-speed cassette loading (1500 and 500 baud). You can expand your Model III to support up to four (two integral) double-density disks at 175K each for a total system capacity of up to 670K bytes.

Radio Shack is Lowering the Cost of High Technology!

As Low As $699*

Powerful Memory! Up to 48K of internal memory is easily added, since no expansion interface is required. Model III is completely self-contained. Start with a 4K Level I system or move up to our 16K Model III BASIC right away for the applications you need.

High-Resolution Display! Every Model III has a sharp display of 16 lines of 64 characters. Model III BASIC adds lower case plus graphics and special characters.

Feature Packed! Every Model III includes a parallel printer interface and 65-key keyboard. Model III BASIC adds "extras" like a real time clock, scroll protect, keyboard controlled screen print, and RS-232 firmware.

And It's Very Affordable! The 4K Level I system is only $699.* The 16K version with powerful Model III BASIC is just $999.* So why wait, step up to the new standard now. Available at Radio Shack stores, dealers and Computer Centers everywhere.

Radio Shack®

The biggest name in little computers™

Send me your TRS-80 Catalog!

Radio Shack, Dept. 81-A-41
1300 One Tandy Center
Fort Worth, Texas 76102

Name
Street
City State
Zip Phone

*Retail prices may vary at individual stores and dealers. Special order may be required initially.

Walter Koetke

COMPUTER BLACKBOARD

Whatever Works

Use of the microcomputer as a "what if" machine for students is a popular notion whose validity has been demonstrated for several years using time-sharing terminals. With very similar capabilities now available on much less expensive microcomputers, many new applications are possible. One of these is to provide a "whatever works" machine for the teacher.

Good teachers have long been aware that providing appropriate motivation is a major part of their job. Unfortunately, those factors that today motivate one student not only may not work on his peers, but they may also not even be useful with him on another day. The microcomputer doesn't provide a motivation that will work with all students at all times, but it does provide the teacher with a motivational tool that can be adapted to a wide variety of situations.

Some years ago I was working with a high-school program that offered six different computer electives using local time-sharing terminals. Although we made an effort to have students of all abilities use the computer facilities, our efforts were only marginally successful. The vast majority of students participating in the electives were among the academically talented.

To counteract this situation, we created a seventh computer elective. The course title was the equivalent of "Fun and Games with the Computer." We could have used Ted Sage's very fine book of the same name at that time. Our new course had a very important prerequisite: Students could only enroll if they had not already taken, were not now taking and probably never would take Algebra 1. This prerequisite eliminated at least 85 percent of the student population.

The initial offering of this course was oversubscribed. About 15 students were expected, and 32 arrived. The computer had provided the motivation we sought. Thirty-two students were voluntarily enrolled in a course and were sitting in a classroom rather than spending the same period of time in the parking lot, in the lavatory or in some other non-supervised location.

With a good gimmick, you can easily draw a crowd. Keeping the crowd's attention and interest is much more difficult. The curriculum material for the new course was now in a do-or-die situation. What was the curriculum for this usually tough-to-manage group of students? The answer was primarily games. The time-sharing library was amply stocked with everything from tic-tac-toe to chess, and students were free to play whatever game, run whatever simulation or just use whatever programs they found in the library. One very defendable rationale for this curriculum was that anything the students did in the computer lab was of more value to them than anything they might have done during the alternative unsupervised free period.

Was the curriculum a complete success? No, three students dropped the course during the first two weeks. However, the results with the remaining 29 students were very encouraging. Before the first week of school was over, one student came in after school to say she didn't just want to "play those games"; she wanted to learn to program.

In one way or another, every student in the class did exactly the same thing before the quarter ended. One boy lasted until the final two days before making his request, but he did ask. The computer had helped a rather difficult group of students take an enormous motivational step as they each went to a teacher and expressed a sincere desire to learn. As any teacher will confirm, when the student says "I want to learn," the battle is over, and a rewarding aspect of education begins.

Don't be afraid to permit the use of games on microcomputers in your school. They can provide several useful support functions, not the least of which is motivation for students who might not be reached in more conventional ways.

The example discussed was accomplished with Teletypes. Today's microcomputers with high-speed CRTs, color, sound and a variety of peripherals can provide far more spectacular motivation. Don't hesitate to use them that way.

Now consider an altogether different situation. A good friend once requested a program for a young man who needed some flash-card-type drill with the multiplication tables. A brief program written for the TRS-80 to accomplish this is illustrated in Listing 1.

The student is provided 25 randomly generated multiplication problems from the desired multiplication tables (0 through 12 in the listing). If a problem is answered correctly, the student receives immediate positive feedback. If a problem is answered incorrectly, the problem is repeated and the correct answer is given. After 25 problems have been attempted, the number of correct answers is indicated and the interaction is complete. Note the use of lines 190 and 200. These allow students to continue at their own rate while still maintaining an uncluttered display on the CRT. The program did almost everything required.

Why is the program only "almost" everything required? The third grader for whom it was written used the program for less than five minutes, then went off to do something else. When asked why he quit so soon, his response was "It's boring!"

The program did everything required except motivate. The young man wasn't motivated to learn the multiplication tables before the program was available, and the program did nothing to change his opinion.

Fortunately, a BASIC program on a microcomputer can be tailored to meet the needs of the user. As the program is written, the student's only reward for answering a question correctly is the word CORRECT. That's certainly interactive, and may even be rewarding the first few times. But how rewarding is the 20th CORRECT? The young man's description was rather accurate. The BASIC responses illustrated in Listing 2 added the missing dimension of motivation for this particular student.

With the addition of these commands, the program selects and prints a random positive comment after each correct answer. In Listing 2 there are 20 comments from which to choose. When actually done with the student being discussed, there were 50 such comments. This modification was a huge success with the previously bored student. His first use of the modified program lasted three hours. He was making a written list of all the different responses he received from the computer. He'd almost forgotten he was doing a multiplication drill. His

```
100 K=0
110 FOR C=1 TO 25
120 P=RND(13)-1 : Q=RND(13)-1
130 CLS : PRINT "PROBLEM" C : PRINT
140 PRINT P "*" Q "= " ;
150 INPUT A
160 PRINT
170 IF P*Q<>A THEN PRINT "NO," P "*" Q "=" P*Q
180 IF P*Q=A  THEN PRINT "CORRECT" : K=K+1
190 PRINT @976, "PRESS THE C-KEY TO CONTINUE" ;
200 IF INKEY$<>"C" THEN 200
210 NEXT C
220 CLS : PRINT @256,"YOU HAD" K "PROBLEMS CORRECT"
230 END
```

Listing 1. Multiplication drill program for the TRS-80.

objective was a complete list of different responses. That he had to answer the arithmetic problem correctly to get a response was only incidental. His teacher, however, was delighted.

The technique of providing random reinforcing comments is often effective, especially if the teacher personalizes the list of possible comments by including those remarks currently popular with the students. For example, today's elementary students find EX-CELLENT or DECENT far more rewarding than COOL or SWIFT. Making a list of 50 or more responses that your students will enjoy is a fun challenge, and one that can make you feel a little dated when students look only puzzled at your favorite expressions.

The programming technique illustrated in Listing 2 is very straightforward and can be adapted in a variety of programs. The PRINT "CORRECT" of line 180 was replaced with GOSUB 300 to minimize changes to the existing lines. Note that line 350, the first DATA item, contains the number of different comments available. If you feel ambitious and make a total of 50 positive comments, then line 350 should read DATA 50. Try adding similar lines to one of the programs you would like students to use. You may need a little help if the program already includes READ/DATA statements. If there are no READ/DATA statements, you won't have to do any more than type the statements given in Listing 2.

Will the technique of randomly selected reinforcing comments work with all students? Of course not. So dig into the capabilities of your "whatever works" machine and try another technique. Although the technique we're about to examine would be expensive for classroom use, it's effective for those with microcomputers at home.

Simply stated, the new technique offers 25¢ every time the students correctly answer 25 consecutive problems. If an error is made before 25 problems are correct, the program terminates. Listing 3 contains a complete program that includes this technique.

Note the revision of line 230, which now prints the motivating message. Note also the programming technique used to stop the program when an error is made. Adding C=25 to the end of line 170 causes the computer to think it has completed the FOR/NEXT loop that is counting problems.

Personal experience with this type of motivation revealed the need for an additional feature. Once a problem is presented, the student must have a limited amount of time in which to respond. If he takes too long, the problem should be counted as incorrect. This feature has been included in the program in Listing 3.

Because timed input can be useful in a variety of situations, take a few minutes to understand the programming techniques required. The INPUT A command of line 150 in Listing 2 was replaced by GOSUB 500 in Listing 3. Lines 500 through 580 are then used as an input subroutine to permit timed numeric input.

The variable T is used to control the amount of time permitted for the student response. By modifying the IF command in line 520, you change the time. By changing IF T=200 to IF T=150, the time is decreased. By changing to IF T=250, the time is increased. Experiment with these values. You can vary the delay to meet the individual needs of each student.

The variable A is used to store the value entered by the student. If the time limit is exceeded, A is given the value −1, which can then be identified elsewhere in the program.

Note that the input subroutine works much like the INPUT A statement it replaced. The student must press the enter key after typing the answer (line 530 in the subroutine), and the left arrow can be used to delete a single character (line 550 in the subroutine).

Some readers may consider the idea of monetary reward inappropriate. If that's the case, don't use it. The program in Listing 3 may still be valuable for some of the programming techniques it illustrates. For those who don't object to this technique, I offer the personal experience of a son who learned his multiplication tables exceedingly well for $3.25. Although my field is not finance, I consider that a very sound, high-yield investment.

I hope the examples in this article have demonstrated three of the "whatever works" possibilities of educational microcomputers. Different students are motivated in many different ways, and the microcomputer is a flexible tool that permits teachers to individualize the presentation format of many ideas. The programming techniques illustrated can be implemented in your own programs as well as those you've purchased and then modified. If the result truly helps a student learn, your efforts will have been worthwhile.

```
180 IF P*Q=A  THEN GOSUB 300 : K=K+1
300 RESTORE : READ N
310 R=RND(N)
320 FOR Y=1 TO R : READ R$ : NEXT Y
330 PRINT R$
340 RETURN
350 DATA 20
360 DATA YOU GOT IT, RIGHT, EXCELLENT, CORRECT, OK, TERRIFIC
370 DATA YES!!!, PERFECT, RIGHT ON, DIRECT HIT, SUPER ANSWER
380 DATA YEA YEA, FANTASTIC, THREE CHEERS, YOU BLEW IT AWAY!
390 DATA POW!, EX-CELL-ENT, WHAMO, YOU MADE IT LOOK EASY
400 DATA HOORAY
```

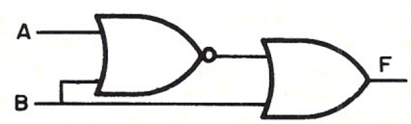
Listing 2. Motivational commands added to Listing 1.

```
100 K=0
110 FOR C=1 TO 25
120 P=RND(13)-1 : Q=RND(13)-1
130 CLS : PRINT "PROBLEM" C : PRINT
140 PRINT P "*" Q "= " ;
150 GOSUB 500
160 PRINT
165 IF A=-1 THEN PRINT "TIMES UP!" : PRINT
170 IF P*Q<>A THEN PRINT "SORRY," P "*" Q "=" P*Q : C=25
180 IF P*Q=A  THEN GOSUB 300 : K=K+1
190 PRINT @976, "PRESS THE C-KEY TO CONTINUE" ;
200 IF INKEY$<>"C" THEN 200
210 NEXT C
220 CLS : PRINT @256,"YOU HAD" K "PROBLEMS CORRECT"
230 IF K=25 THEN PRINT @512,"CALL YOUR FATHER -- HE OWES YOU A QUARTER!"
240 END
300 RESTORE : READ N
310 R=RND(N)
320 FOR Y=1 TO R : READ R$ : NEXT Y
330 PRINT R$
340 RETURN
350 DATA 20
360 DATA YOU GOT IT, RIGHT, EXCELLENT, CORRECT, OK, TERRIFIC
370 DATA YES!!!, PERFECT, RIGHT ON, DIRECT HIT, SUPER ANSWER
380 DATA YEA YEA, FANTASTIC, THREE CHEERS, YOU BLEW IT AWAY!
390 DATA POW!, EX-CELL-ENT, WHAMO, YOU MADE IT LOOK EASY
400 DATA HOORAY
500 T=0 : A=0
510 T=T+1 : IF T=200 THEN A=-1 : GOTO 570
520 X$=INKEY$ : IF X$="" THEN 510
530 IF X$=CHR$(13) THEN 570
540 PRINT X$;
550 IF X$=CHR$(8) THEN A=INT(A/10) : GOTO 510
560 A=10*A+VAL(X$) : GOTO 510
570 PRINT
580 RETURN
```

Listing 3. Earn while you learn program.

MICRO QUIZ

Find all ordered pairs (A,B) which make F true.



Answer on page 212.

Robert W. Baker

PET-POURRI

D & R Tape Fix

Back in the May column, I reviewed a cassette system from D & R Creative Systems, PO Box 402, St. Clair Shores, MI 48080, that used a Sanyo recorder with a built-in counter. I mentioned then that the only disadvantage I could see with their system was that the microphone and ear cables to the recorder could not be connected at the same time. Whenever you want to switch between reading or writing a tape, you have to switch the cables to the recorder.

A recent letter from D & R Creative Systems outlined a simple fix for this problem: remove one resistor from the recorder circuit board. The diagram in Fig. 1 shows the location of the 47 ohm resistor that must be removed.

This change has been incorporated in all units delivered after July 1. Since Sanyo has dropped their model M2545A recorder with the fast forward cueing feature, D & R is replacing it with the M2544A model without the cueing feature.

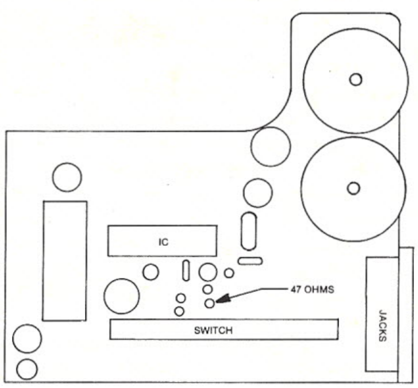


Fig. 1.

TNW Products

TNW Corporation (3351 Hancock St., San Diego, CA 92110) offers several serial interfaces designed to interface RS-232 devices to the Commodore PET/CBM and other IEEE-488 bus computers. Each unit can share the IEEE bus with other peripheral devices, and each provides a PET-style bus connector for daisy chaining. All units include a power supply and cable for use with the PET (TNW-1000 mounts directly on the PET), and each provides old PET/ASCII lowercase conversion. Table 1 compares several models and their features.

For more advanced applications, TNW also offers a low-speed modem for $389. The TNW 488/103 is a frequency shift keyed (FSK) modem compatible with the Bell 103 modem.

	TNW-232D	TNW-2000	TNW-1000
RS-232 serial outputs	2	1	1
RS-232 serial inputs	2	1	1
Software accessible RS-232 control signals	6 input and 6 output	0	0
Current loop device supported	with adapter	with adapter	yes
RS-232 connectors	1 male & 1 female $35 option	female included male is a $25 option	
New PET/ASCII lowercase conversion	compatible with Word Pro others require software		yes
IEEE bus address range	0–27	0–15	4–7, 12–15
IEEE bus addresses required	4	1	1
Cabinet included	yes	yes	no
List price	$369	$229	$129

Table 1.

Low speed means up to 600 bits per second (baud), but most systems run at only 300 or 110 bits per second.

The TNW 488/103 connects directly to the telephone network via a separate data access arrangement (DAA), not an acoustic coupler. Your computer can dial the telephone for you and answer when other computers call. You can purchase the DAA for $159 or rent one from the telephone company for about $6 per month.

Included with the TNW 488/103 is a program called PTERM that allows you to use your PET as a standard ASCII CRT terminal. The program properly handles conversion between the PET and ASCII character sets for both new- and old-style PETs. Since the PET does not have a control key, control characters are transmitted by hitting the reverse key and the appropriate character. You can also switch from full- to half-duplex operations and enable or disable output to a printer or disk.

PTERM can be purchased separately for $19, and a version is available for the TNW-232D or TNW-2000 RS-232 serial interfaces.

Another interesting program available from TNW is called SWAP ($19). This system utility program allows several BASIC programs to reside in a PET's memory at the same time. This lets you run multiple programs without having to load from tape between executions.

After loading and running SWAP, you enter the number of separate program areas to be created and allocate the memory space available to each area in 256-byte blocks. Following the initialization, you can activate any program by executing the command ?USR(n), where n is the desired program number. The load command is used to load a program into the currently active area; the run command runs the program in the active area.

Program swapping is performed by a machine-language program that resides in the PET's second cassette buffer. This places limitations on the use of machine-language programs with SWAP; for example, SWAP preempts the USR function. In addition, the swapping process clears variable storage, so that programs in different areas can't be linked. Once a program has been swapped out and then back in again, it can only be rerun and not continued.

CMC Interfaces for the PET

Connecticut Microcomputer, Inc. (CMC), has produced a new, condensed 20-page catalog describing the firm's expanding line of microcomputer interfaces, data acquisition modules and accessories. The CMC interfaces let your system read and measure a variety of real-world variables. Products covered in the literature include the AIM16 A/D converter, BSR X-10 remote controller computer interface, addressable PET printer adapters, Xpandr I simultaneous multiple-input connectors, Tempsens dual temperature probe and a variety of connectors.

The CMC AIM16 is a 16-channel analog-to-digital converter that is connected to the host computer via an eight-bit input port and an eight-bit output port, or through one of CMC's custom interfaces (PETMOD for the PET). The input voltage is converted to a count between 0 and 255 (00 and FF hex). Resolution is 20 millivolts per count, with an accuracy of .5 percent, plus or minus one bit. Conversion time is less than 100 microseconds per channel, and all 16 channels can be scanned in less than 1.5 milliseconds. The compact module sells for $179 and requires an external 12 V dc, 60 mA power source. Power supplies are available at $14.95 and $24.95, depending on the desired input line voltage.

SADI, CMC's new addressable PET printer adapter, is a microprocessor-based serial and parallel interface. It lets you connect the PET to parallel and serial printers, CRTs, modems, acoustic couplers, hard-copy terminals and other computers. The serial and parallel ports are independent, allowing the PET to communicate with both peripheral devices simultaneously or one at a time.

Special features for the PET interface include:

- Conversion to true ASCII (both in and out),
- Cursor controls and function characters specially printed,
- Selectable reversal of uppercase and lowercase,
- PET IEEE connector for daisy chaining and
- Full addressability—works with other IEEE or Commodore devices.

Special features for the serial interface include:

- Baud rate selectable from 75 to 19200,
- Half or full duplex,
- 32-character buffer,
- X-on, X-off automatically sent and
- Selectable carriage return delay.

Special features for the parallel interface include:

- Data strobe, either polarity, and
- Device ready, either polarity.

SADI sells for $295, fully assembled with power supply, PET-to-IEEE cable, RS-232 connector, parallel port connector and a case. An addressable RS-232-only interface, the ADA 1400, is also available at $179 and includes several printer utility programs on tape.

All CMC products are available from the factory and from many local dealers. For more information or a copy of CMC's latest catalog, write Connecticut Microcomputer, Inc., 34 Del Mar Drive, Brookfield, CT 06804.

MAE User's Group

A new user's group has been formed for Eastern House Software's MAE macro assembler reviewed in the August column. Currently it is operating much like the early *PET Gazette* exchange, with users contributing programs and getting other programs at minimal cost. They may even issue an occasional newsletter. The goal is to exchange programs among all 6502-based machines, since many programs will work on all with simple changes.

I just received two full disks of various utility programs from the exchange but haven't had time to try many. Included on the disks were copies of Extramon, an extended monitor; BASIC Aid, an extension of BASIC, adding many convenient debugging and editing features such as renumbering, auto line numbering, tracing, find and change functions and block deleting; EPROM programmer software; a basic word processor, with more enhanced versions to be possibly added later; various four-part music programs; symbolic disassembler; and various useful disk utilities.

For now, copies of the utility disks are $10 per disk, if you supply the disk. Individual listings are $2 each. Another utility disk, primarily for MAE 4.0 users, is currently being readied. For more information, contact James Strasma, c/o Grace UMC, 120 West King St., Decatur, IL 62521.

NEECO Source Kit

New England Electronics has a complete package for connecting your PET/CBM to STC's Source Information Utility. The Source provides access to *New York Times* news service, UPI stock reports and much more. It allows programming in FORTRAN, COBOL, RPG, assembly or extended BASIC. A number of entertainment programs such as Adventure and Star Trek are also available, along with an electronic mail system. The NEECO Source kit includes a communications interface cable and a smart terminal software package.

The interface cable converts the output of the PET parallel user port to a compatible RS-232 output. This interface lets you connect directly to most RS-232 standard acoustic coupler modems without any expensive hardware interface.

The software package is based on Alpha Software's Intelcom. This program lets the PET pass your Source account number and password to the timeshare mainframe with only two keystrokes. From that point on, you are on The Source. You can capture files from The Source and save them on disk. The program also gives you the capability to later pass those files to your printer or save them under a permanent name for later processing. You can even create your own files on disk and pass these to The Source.

Price of the complete Source terminal package is $99.95, which includes software for cassette-based systems that do not have a 2040 disk available. This option does not include the capability to pass files in or out. The terminal package may be used with timesharing systems other than The Source, but you must manually enter your access number, passwords and any

```
10 REM      WIND CHILL TEMPERATURE
20 REM
30 REM        BY - ROBERT BAKER
40 :
50 PRINT"▒▒▒▒"TAB(10)"W I N D   C H I L L"
60 DIM C(8,11)
70 FOR W=0 TO 8 :FOR T=0 TO 11
80 READ C(W,T) :NEXT T :NEXT W
90 DATA-60,-50,-40,-30,-20,-10,0,10,20,30,40,50
100 DATA-68,-57,-47,-36,-26,-15,-5,6,16,27,37,48
110 DATA-95,-83,-70,-58,-46,-33,-21,-9,4,16,28,40
120 DATA-112,-99,-85,-72,-58,-45,-36,-18,-5,11,22,36
130 DATA-124,-110,-96,-82,-67,-53,-39,-25,-10,3,18,32
140 DATA-133,-118,-104,-88,-74,-59,-44,-29,-15,0,16,30
150 DATA-140,-125,-109,-94,-79,-63,-48,-33,-18,-2,13,28
160 DATA-145,-129,-113,-98,-82,-67,-49,-35,-20,-4,11,27
170 DATA-148,-132,-116,-100,-85,-69,-53,-37,-21,-6,10,26
180 PRINT"▒▒TEMPERATURE (DEGRESS-F, 50 TO -60)";
190 INPUT T
200 IF T>50 THEN 220
210 IF T>=-60 THEN 240
220 PRINT"▒▒TEMPERATURE IS OUT OF RANGE!"
230 GOTO 180
240 PRINT"▒WIND SPEED (MPH)";
250 INPUT W
260 IF W>=0 THEN 290
270 PRINT"▒▒WIND SPEED CAN'T BE NEGATIVE!"
280 GOTO 240
290 T1=INT((T+60)/10)
300 IF W>40 THEN W=40
310 W1=INT(W/5)
320 A=C(W1,T1)
330 IFT/10=INT(T/10) THEN 460
340 X=C(W1,T1+1)-A
350 D=(T/10)-INT(T/10)
360 A=A+(X*D)
370 IF W/5=INT(W/5) THEN 500
380 A1=C(W1+1,T1)
390 X=C(W1+1,T1+1)-A1
400 D=(T/10)-INT(T/10)
410 A1=A1+(X*D)
420 D=(W/5)-INT(W/5)
430 X=A-A1
440 A=A-(X*D)
450 GOTO 500
460 IF W/5=INT(W/5) THEN 500
470 X=C(W1+1,T1)-A
480 D=(W/5)-INT(W/5)
490 A=A+(X*D)
500 A=INT(A)
510 PRINT"▒▒APPROXIMATE WIND-CHILL TEMPERATURE ="
520 PRINT:PRINTTAB(20);A;"  DEGREES-F"
530 IFA<=-25 THEN PRINT"▒▒DANGER FROM FREEZING OF EXPOSED FLESH!"
540 GOTO 180
READY.
```

Listing 1.

other data required by the particular system sign-on procedures. The Source enrollment fee is $100, and hourly connection charges range from $2.75 per hour (during off hours) to $15 per hour (during business hours).

For more information, write: New England Electronics, 679 Highland Ave., Needham, MA 02194.

PIE

Lem Data Products (PO Box 1080, Columbia, MD 21044) is selling a parallel interfacing element (PIE) that allows connecting any parallel input printer to the PET using the IEEE bus. The PIE has selectable addressing and provides extension of the IEEE-488 bus to be compatible with all other peripherals.

An external +5 V supply is required, but power can normally be supplied by most printers. The PIE provides eight latched TTL data bits and two TTL handshaking lines. Both positive and negative handshaking are supported, so any parallel input device can be driven. An optional, switch selectable code converter ROM will output the correct ASCII codes to match all the ASCII characters displayed on the PET screen.

The PIE sells for $89.95, and the code converter ROM is an additional $14.95. Fully assembled cables for most printers are available for $39.95.

Wind Chill

With winter coming on, the useful little program in Listing 1 will be fun to use. It computes the approximate wind chill temperature from a still air temperature and the wind speed.

To keep things simple, the program uses a table of known values for various temperatures and wind speeds. When you enter a temperature or wind speed that falls between entries in the table, the program simply extrapolates the wind chill temperature for the values entered.

Updates

While talking with Bob Locke of *Compute* magazine recently, I learned that *Compute II* will soon be merged back into *Compute* magazine. The resulting magazine, covering all 6502-based machines, is slated to be published monthly starting in January.

The Paper recently announced the end of its publication due to the editor's illness. However, Ralph Bressler and the Long Island PET Society (LIPS), who assumed publication of *The Paper* back in August, will deliver the ten issues of volume three to the subscribers. New subscribers can get these same ten issues for $15. The new address is *The Paper*, Box 524, E. Setauket, NY 11733.

A new computer club is forming in Rhode Island for owners of Commodore PET/CBM computers. The PET Information Exchange plans to publish about 15 newsletters during the year, and they are currently working on several interesting club projects. Dues are $6 per year. For more information, contact Scott Summer, 27 Leicester Way, Pawtucket, RI 02860.

Several months ago, a professor from a Canadian university sent a copy of a program he had purchased but whose documentation he was having trouble understanding. The program was a multiple regression analysis program for the PET called PRO-GRESS. It sells for $50 from Cognitive Products in Chapel Hill, NC.

The program itself appears well written, and the documentation is rather extensive. However, there were no sample data sets or clear examples on how to use the program. Considering the complexity of the material involved, the documentation would seem to be very confusing to anyone not already familiar with the material. After playing with the program for some time, I'm still lost on how to use it. If anyone has used the program successfully, I'd appreciate a quick note.

In my June review of PET Pilot, I failed to mention that the PET Pilot Editor program requires *two* cassette drives. I hope this didn't cause any problems. Commodore is now distributing PET Pilot, which should be available through most PET dealers.

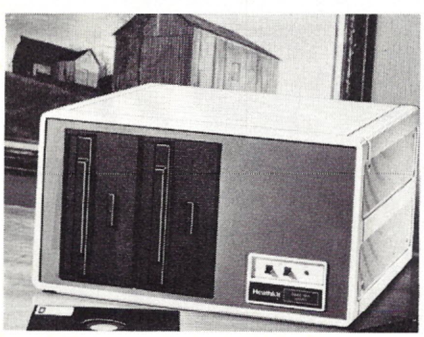
Edited by Dennis Brisson

NEW PRODUCTS

Heath's New Floppy Disk System

The H47 is a new floppy disk system with two-megabyte storage capacity from Heath Company, Benton Harbor, MI 49022. This eight-inch, dual-sided, dual-density floppy disk system, designed for use with Heath's H8 and H89 microcomputers, provides up to 2 1/2 million bytes of on-line data storage. Access time averages 176 ms. The H47 is fully compatible with current Heath 5 1/4-inch disk systems. Both Heath's HDOS Operating System and CP/M permit transfer of data between 5 1/4- and eight-inch disks.

Disk boards, providing interfacing between the H8 or H89 and the H47, are offered separately. A 40-conductor flat cable is included with the H47 to connect the floppy disk system with the H8 and H89 interfaces. Panel switches are included and allow write-protection for each drive, if desired. Reader Service number 480.

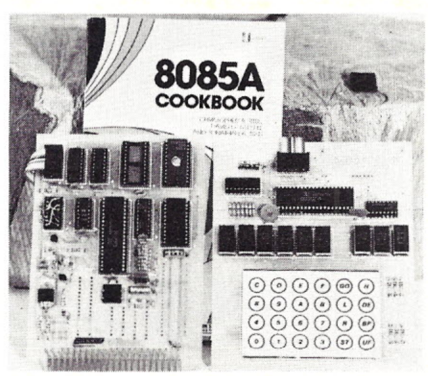


Heath's H47 dual-drive floppy disk system.

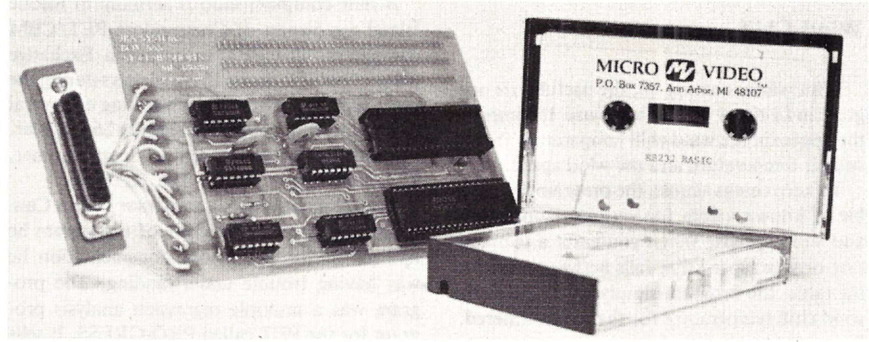
8085A Microprocessor Training Unit

The 8085AAT Microprocessor Training Unit (MTU) includes an 8085A microcomputer with 1K RAM, 1K PROM and 1K EPROM memory, programmable I/O, keyboard unit, CPU card, display and operatory system and a 20 mA asynchronous port. Its 44-pin edge connector allows configuration to any bus structure, and it includes an area on the CPU card for custom wire-wrap design or user-defined interface circuitry. It is completely expandable.

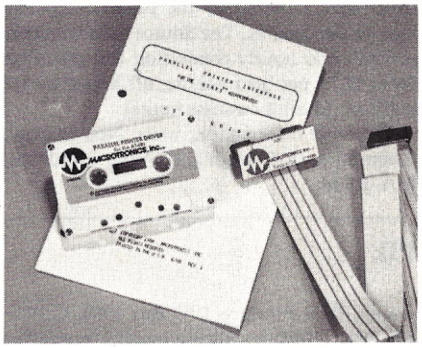
The MTU software includes an instruction manual; a user's manual; the *8085A Cookbook*, which ranges from basic microprocessor concepts to actual design of an 8085A microcomputer; and a software design book with



Paccom's Microprocessor Training Unit.



Micro Video's RS-232 Pack.



Macrotronics' interface package for Atari.

over 190 executable program examples, plus detailed examination of all 244 instructions and typical assembly language for the 8080/8085A microprocessor. Price is $299.95 ($249.95, kit).

Paccom, 14905 N.E. 40th St., Redmond, WA 98052. Reader Service number 487.

RS-232C Peripheral Interface

Line printer and communications access are now possible for Interact computer owners with the RS-232C peripheral interface package from Micro Video, PO Box 7357, 204 E. Washington St., Ann Arbor, MI 48107.

The interface is equipped with a dual port that has handshaking and send/received capabilities for driving any RS-232-compatible device. Installation requires no soldering. The port's design features low-power, trouble-free operation and upward compatibility with future hardware and software enhancements. All I/O parameters are software-selectable from BASIC or machine code. The RS-232 Pack includes Microsoft BASIC with printer access commands and a BASIC editor. Price is $129.95. Reader Service number 490.

32K 6809 System

Gimix's 6809 systems feature a 25 amp constant-voltage ferroresonant transformer, fifteen 50- and eight 30-pin bus slots, a minimum of 32K of static RAM and a choice of I/O cards. A variety of system monitor options, including the GMXBUG 09 monitor/debugger and SWTP's SBUG-E monitor, are available.

The 6809 CPU SS-50 processor board features selectable processor clock speeds of 1, 1.5 and 2 MHz. It has provisions for a variety of onboard devices, including a 9511 or 9512 arithmetic processor, 6840 programmable timer, time of day clock with battery back-up, 1K of scratchpad RAM and four PROM/ROM/RAM sockets that can hold up to 32K of onboard software. Memory management options available include Gimix's enhanced dynamic address translator (DAT) and an SWTP-compatible DAT. Extended addressing allows the processor to address up to 1 megabyte of memory space. Prices start from $1844.69.

Gimix, Inc., 1337 West 37th Place, Chicago, IL 60609. Reader Service number 488.

Digital Logic Probe And Logic Pulser

OK Machine and Tool Corp., 3455 Conner St., Bronx, NY 10475, has introduced two new products for circuit troubleshooting and testing.

The PRB-1 digital logic probe detects pulses as short as 10 ns with frequency response better than 50 MHz and automatic pulse stretching to 50 ns (+ and −). The PRB-1 is fully compatible with all RTL, DTL, HTL, TTL, MOS, CMOS and microprocessor logic families. It also features 120k ohm impedance, power lead reversal protection and overvoltage protection to 200 V. Price is $36.95.

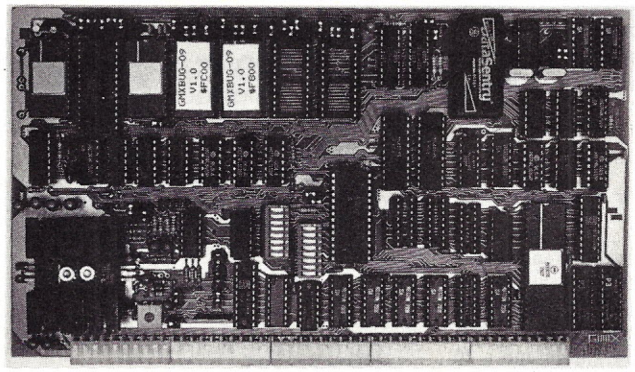
The PLS-1 logic pulser will superimpose a dynamic pulse train (20 pps) or a single pulse onto the circuit node under test. There is no need to unsolder pins or cut printed-circuit traces even when these nodes are being clamped by digital outputs. This multi-mode, high current pulse generator can source or sink sufficient current to force saturated output transistors in digital circuits into the opposite logic state. Signal injection is by means of a pushbutton switch near the probe tip. Price is $48.95. Reader Service number 485.

S-100 Video Graphics Board

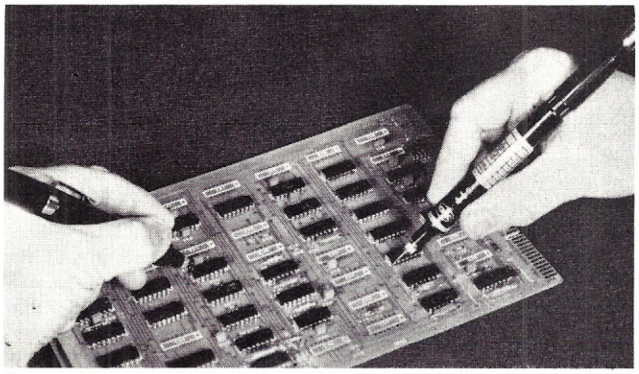
Primarius, Inc., 4186-J, Sorrento Valley Blvd., San Diego, CA 92121, offers an onboard, dual port, 6K byte video RAM for the S-100. It uses the Motorola MC6847. Alpha, semigraphics and full graphics modes are I/O selectable. The design implements the memory wait technique to allow concurrent access of video RAM by both the CPU and the video chip. This allows flicker-free video update during scan time. The entire screen can be updated in less than 60 ms for realistic animation. Price is $250. Reader Service number 475.

Printer Interface for Atari

A parallel printer interface for the Atari microcomputers is now available from Macro-



The Gimix 6809 CPU board.



OK Machine's PRB-1 and PLS-1.

META TECHNOLOGIES

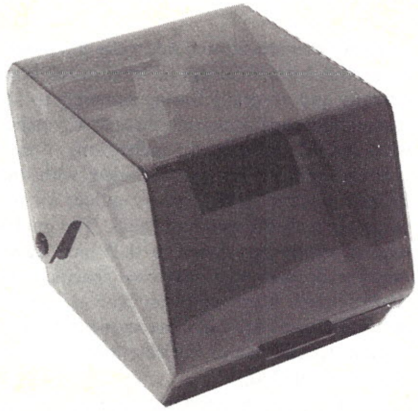
FOR YOUR DISK SYSTEM

FILE BOX

DISKETTE STORAGE SYSTEM

$19.80★ for 5¼" disks

for 8" disks . . . $24.95★



MTC brings you the ULTIMATE diskette storage system, at an affordable price. Storing 50 to 60 diskettes, this durable, smoke-colored acrylic unit provides easy access through the use of index dividers and adjustable tabs. Unique lid design provides dust-free protection and doubles as a carrying handle.

PLASTIC LIBRARY CASES

(not shown)

An economical form of storage for 10 to 15 diskettes, and is suitable for your bookshelf! Case opens into a vertical holder for easy access.

5¼-inch or 8-inch diskette case . . . . $3.00★

Single Sided, Single Density, Soft-Sectored 5¼-inch, (for TRS-80™) Mini-floppy

DISKETTES

$19.80★ box of 10

Meta Technologies strikes again . . . at the competition! These are factory fresh, absolutely first quality (no seconds!) mini-floppies. They are complete with envelopes, labels and write-protect tabs in a shrink-wrapped box.

INTRODUCING PLAIN JANE™ DISKETTES

The Beautiful Floppy with the Magnetic Personality™

In 1980 alone, MTC has sold nearly a third of a million dollars worth of brand-name diskettes. If anyone knows quality, we do. And these are quality diskettes. The catch? They are in a plain white box. You're not paying for fancy printing, fancy labels or fancy names on the packaging. We don't even put our own label on the package (labels cost money). At this introductory price (our regular price will be $21.95 per box of 10) we cannot offer quantity or dealer discounts.

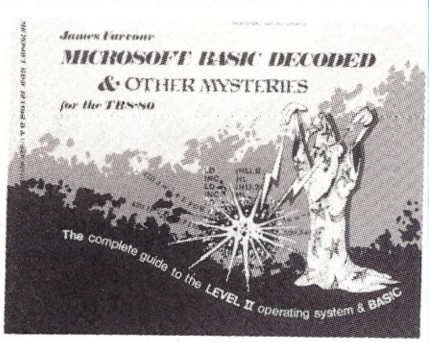
PLAIN JANE™ Diskettes . . . . . . $19.80★

VERBATIM brand Diskettes (box of 10)

5¼-inch (for TRS-80™)
MD525-01 . . . . $23.95
10 boxes of 10 . . . (each box) . . . $22.95

8-inch FLOPPIES
Single-Density, FD34-1000 . . . $29.95
Double-Density, FD34-8000 . . $39.95

TRS-80™ PRODUCTS



NEWDOS/80 by Apparat $149.95
NEWDOS + with ALL UTILITIES
35-track $69.95
40-track $79.95
TRS-80™ DISK AND OTHER MYSTERIES . . . $19.95
MICROSOFT ™ BASIC DECODED & OTHER MYSTERIES for the TRS-80™ $29.95

THANK YOU for 1980 . . .

To show our appreciation for a very successful year, thanks to our thousands of satisfied customers, we are offering some great products at prices even lower than our regular low prices. We make an honest effort to deliver the best products at the lowest prices with the fastest service. The confidence you have shown in MTC is recognized and welcomed. Our mothers thank you, our fathers thank you, our children thank you . . . and we thank you.

CALL FOR INFORMATION ON OTHER TRS-80™ PRODUCTS

All products guaranteed for replacement only. Prices, Specifications & Offerings subject to change without notice.

MOST ORDERS SHIPPED WITHIN ONE BUSINESS DAY

★PRICES GOOD THRU NOVEMBER 30, 1980. Sorry, no dealer or quantity discounts. Allow for shipping delays for 1980 specials.

WE ACCEPT
- VISA
- MASTER CHARGE
- CHECKS
- MONEY ORDERS
- C.O.D.

- Add $2.50 for standard UPS shipping & handling
- $2.00 EXTRA for C.O.D.
- Ohio residents add 5½% sales tax.

TO ORDER CALL TOLL FREE

1-800-321-3552

IN OHIO call (216)289-7500 (COLLECT)

✓161

META TECHNOLOGIES CORPORATION

26111 Brush Avenue, Euclid, Ohio 44132

master charge

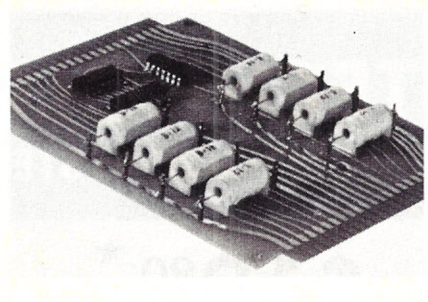
VISA

800917

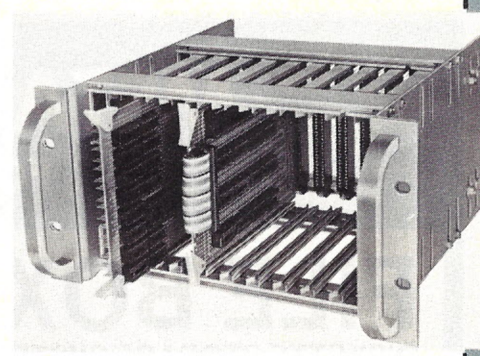
TRS-80 is a TM of Tandy Corp.
PLAIN JANE is a TM of MTC.

tronics, 1125 N. Golden State Blvd., Turlock, CA 95380. It allows the Atari 400 or Atari 800 to directly drive a parallel ASCII printer.

The interface package includes a cable assembly and parallel printer driver on cassette. The interface will drive most seven-bit ASCII parallel printers with handshaking (data strobe and busy signals). Connectors are available for most of the popular printers, giving plug-in installation. Information is provided to connect to almost any other parallel printer. The A4P is for the Atari 400; the A8P fits the Atari 800. Price is $69.95. Reader Service number 481.



Atec's matrix/multiplexer interface module.



Vector Electronic's CCK card cage.

Analog Interface Switching Modules

Atec Systems, PO Box 128, Mendon, NY 14506, has introduced a series of switching modules which can be used as an analog interface between any microprocessor eight-bit I/O port and signals to be switched in automatic test equipment, instrumentation and control system applications.

The modules can be operated from the microprocessor in either a matrix mode, where any switch selected can be latched or unlatched, or a multiplexer mode, where only one switch can be closed at any time. A clear command unlatches all switches in either mode of operation. The latches are solid state, operating at microprocessor speeds; and the switches are sealed reed relays, closing in less than 1 ms and having a life of more than 100 million operations. The modules are 4.5 by five inch circuit boards that can be assembled into large arrays by plugging into prewired card cages in the required configuration. Also, by selecting the required interface module, the complete matrix or multiplexer can be controlled from either an eight-bit I/O port or from the IEEE-488 bus. Reader Service number 489.

10/19-Inch Wide Card Cages

Now designers have a choice between standard 19-inch rack and 10.25-inch "one-half" rack mounting with the CCK Vector-Pak series of four card cages. The cages are 5.25 inches high by nine inches deep and either ten inches or 19 inches wide. Slotted side walls and brackets permit cross members to be adjusted both laterally and vertically during assembly to accommodate card sizes from three to 4.5 inches wide and 4.5 inches to 6.5 inches long. Nylon snap-in card guides are included with the 19-inch racks to hold 21 cards; guides for ten cards are with the one-half racks. The guides may be easily positioned in 0.25-inch increments to accommodate any card spacing. The cages feature ruled scales on both connector mounting struts for fast connector-positioning without special jigs or intricate measurement. Price is about $40.

Vector Electronic Company, Inc., 12460 Gladstone Ave., Sylmar, CA 91342. Reader Service number 479.

Edited by Dennis Brisson

NEW SOFTWARE

Flight with Apple II

Now you can fly your Apple II with the A2-FS1 Flight Simulation program from Sublogic Distribution Corp., Box V, Savoy, IL 61874. The system offers flight simulation that considers 23 aircraft characteristics, a three-dimensional view of the ground and sky, complete flight controls and 18 instrument indicators. The 3D display is like looking through the windshield of a plane. As you roll and bank, the ground tilts accordingly, and as you dive, the ground fills the screen.

The program is written in protected machine code (i.e., it cannot be copied). As you load the program, it loads its own loading bootstrap and then the program itself. According to the instruction manual, Sublogic will replace the tape if you have any loading problems.

I used the cassette version, which loads extremely well. It came right up on the first try, and seems to be relatively stable volume-wise. Loading the program takes 90 seconds and requires the full 16K memory. Because the program is written in machine code, it runs very fast and is capable of updating the 3D display as fast as five times per second. This gives a smooth display, without much flicker.

Once the program is loaded, it will take off running by itself. As it begins, you find yourself on the refuelling ramp of a WWI British air base. After becoming familiar with your controls and flight maps, pour on the throttle to exceed 60 mph. As you look out of your windscreen and see the ground drop away, you know you are flying. Once in flight, the program takes into account many factors, such as lift, pull of gravity, drag and stalls, to determine your plane's performance.

Once you have mastered the art of flying, you can test your aviational skills in a war game called British Ace. Your mission is to bomb an enemy fuel supply depot, while warding off the attack from five enemy planes. The program will support game paddles, joysticks or even keyboard input on the Apple. Sublogic's flight simulation program is available for the Apple, as well as the TRS-80, on cassette for $25. It is also available for the Apple on disk for $33.50. Reader Service number 494.

Scott King
New Hope, MN

Flight simulation for Apple II.

FORTH

The FORTH language, with its fast operating speed and increased usage in microcomputer applications such as graphics, robotics, process control and telecommunications, has recently become available for several systems.

Eric C. Rehnke Tech Services, 1067 Jadestone Lane, Corona, CA 91720, has announced the availability of the FORTH programming system for the 6502-based KIM-1, SYM-1 and AIM-1. This version of FORTH contains a built-in 6502 assembler, a text editor and a cassette file management system. Information on interfacing FORTH to a floppy disk and several extensions to the language are also provided. Price is $90. Reader Service number 497.

FORTH for the Apple II is available from Cap'n Software, PO Box 575, San Francisco, CA 94101. This version 1.7 includes the FORTH Interest Group programming language plus extensive development aids and a 130-page tutorial manual. It also includes a structured macro assembler, which allows you to create machine-language subroutines, which are immediately ready to run when entered, saving development time. A screen editor, graphics and other Apple utilities are included. The system runs on Apple II, Apple II + or Apple II with language card; one or two disks; and 48K memory. Price is $140. Reader Service number 498.

FORTH for CP/M is available from Mitchell E. Timin Engineering Co., 9575 Genesee Ave., Suite E-2, San Diego, CA 92121. FIG FORTH is supplied on an eight-inch, single-density diskette and requires at least 24K. A FORTH-style editor with 20 commands, as well as a virtual memory subsystem for disk I/O, is included. Other features include a Z-80/8080 assembler and an interleaved disk format that minimizes the time required for disk access. Price is $75 for the eight-inch format and $90 for other diskette formats. Reader Service number 499.

Stock Market Monitor

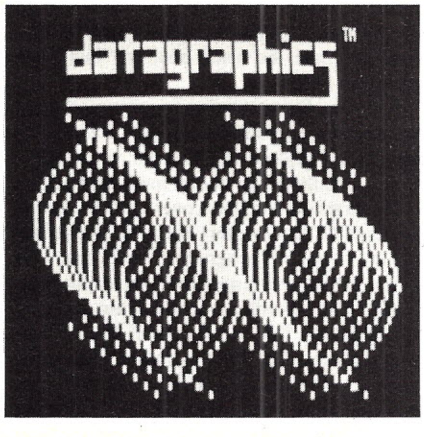
The Stock Market Monitor System, designed for the active trader, rather than the long-term investor, tracks user-selected issues to discover the issue's performance against the overall market. Set-up data is input by the user from the *Standard and Poors* stock guide or *Value Line*. Daily issue data (high, low, close and volume) is input from any newspaper containing this information. Daily overall market volume and closing Dow are also provided from a newspaper.

The system's analysis of a given issue is done by comparing volume and price changes of the issue to volume and price changes of the overall market. From these comparisons, you may determine whether the issue is outperforming, under-performing or performing with the market. The system also performs comparisons of the issue against itself. Designed for the TRS-80 Model I, Level II (16K or more), it is available on cassette ($89) and disk ($99).

Galactic Software Ltd., 11520 N. Port Washington Rd., Mequon, WI 53092. Reader Service number 483.

Curves

Datagraphics, PO Box 566, Dept. G, Union Station, Endicott, NY 13760, offers its first in a series of programs on graphics applications programming techniques for the TRS-80. The first course, Curves, plots curves using a simple arithmetic progression/regression technique that allows displays to be realized on the video monitor in five to 20 seconds. The program starts with a simple explanation of For-Next loops and line numbers; continues with amplitude equations, regressions and progressions; and ends with a program of various designs for designing computer art. Price is about $20 for the 16K Level II or 4K Level I cassette. Reader Service number 492.



Sample output from Datagraphics' Curves program.

Genealogy

Genealogy is an application subsystem that lets you trace not only the usual genealogy information—who your parents are, and their parents, etc.—but also the entire multiple generation family, including cousins ten times removed. Written in North Star BASIC, the system requires an 80-column character or matrix printer. Its data-base programs consist of a name file and a detailed information file that contains a record for each person identified. The records are chained to each other by multiple linkages that reflect the various relationships that exist between people. A single density, single-sided mini diskette will enable you to build a data base with 250 members. Diskette price is $45.

The program generates reports that allow you to highlight the birth, anniversary and memorial dates to remember on a monthly basis; trace the bloodline of a selected individual; and print a selected individual's ancestral heritage for six generations.

Bio-Charts Co., PO Box 423, Nanuet, NY 10954. Reader Service number 491.

Word/Data Processing System

The T/Maker system combines word processing and data processing for 8080/Z-80 microcomputers to provide analysis and presentation of numerical data and text copy used in financial modeling and report preparation. Typical T/Maker applications include sales projections, profitability studies, balance sheets, estimates and price sheets.

T/Maker requires a 48K CP/M system and CBASIC-2. The system includes a full screen editor for word processing and report generation. Computation for rows and columns includes standard arithmetic, percents, exponents, common transcendental functions, averages, maxima, minima and projections. With its visual two-dimensional syntax for computing tables and other features, creating, modifying and restructuring tables become as easy as entering the data. Files can be inserted, appended and sorted. Data files can be created, loaded and processed automatically. Price is $275.

Lifeboat Associates, 1651 Third Avenue, New York, NY 10028. Reader Service number 476.

Appointment Calendar

The Appointment Calendar, from Charles Mann & Associates, Micro Software Division, 7594 San Remo Trail, Yucca Valley, CA 92284, can handle office receptionist functions for single-practice practitioners and group service operations. The system can schedule up to 19,000 appointments per client group (each group containing up to 10,000 active clients).

The program allows the receptionist to create temporary and permanent client files and to schedule any length appointment either on the phone or at the office. A simple keystroke selects the appointment, enters it onto the daily appointment log and prints a mailable appointment notice for the client. It provides for set break and lunch periods and blocks out such non-service days as vacations and holidays. An on-screen HELP facility is provided. The system is designed for the Apple II or Apple II Plus computer and requires 48K RAM, at least two mini-disk drives and an 80-column printer. Price is $189.95. Reader Service number 493.

1980 Tax Program

A new tax help will be available for TRS-80 owners in the preparation of their 1980 income taxes. Tax/Saver helps you prepare taxes in a professional manner according to the latest tax rules. If there is more than one way of doing the return, the program lets you compare and choose the best result. Applicable for both the long and short forms, Tax/Saver compares itemized deductions to national averages, computes medical deductions and contributions, handles community property, checks for excess FICA and helps determine dependents. It is available on cassette for the 16K, Level II, for

$65, and on disk for the 32K with two disk drives for $80.

Micromatic Programming Co., PO Box 158, Georgetown, CT 06829. Reader Service number 496.

Apple II ACES Simulation Program

The ACES (Apple II Continuous Equation Simulator) program provides large differential equation simulations for use in education and engineering areas such as control system, electronic circuit, aerodynamic, thermodynamic and fluidic analyses. It is written in Applesoft and allows interactive run/rerun features. Solution outputs are provided via a high-resolution graphics plot and a screen/printer tabular listing. The program allows a Disk II system to be effectively utilized in storing output solution plots. Simulation problem size can be in excess of 150 integrators on a 48K system with DOS overhead. Price is $149.95.

Modulo 2 Company, PO Box 3795, University Park, NM 88003. Reader Service number 477.

Microcommunicator

The Microcommunicator can transform your Apple II or Apple II Plus into a communications device for the severely physically disabled who cannot speak, write or type. A single keystroke by finger or mouthstick will display any sentence chosen from 60 or more programmed sentences, which can be changed by the user at any time. Messages of up to 100 words and phrases can be constructed for display or printout (optional) by double keystroke selections of a built-in vocabulary that exceeds 1600 sentence-building words, phrases and suffixes. The system, which requires a single-disk drive and monitor, is available with adult or children's vocabulary. Price is about $40.

Grover & Associates, Creekside Center, Suite D 116, 7 Mount Lassen Drive, San Rafael, CA 94903. Reader Service number 495.

Computer Tutor

Computer Tutor is an educational software package that presents questions in a random order, with the correct answers appearing in a different position each time. The package consists of the following topics—geography of the world, geography of the United States, commodities of the world, commodities of the United States and United States government. Each series is accompanied by a number of charts referenced by the program, enabling you to follow along with printed information. Correct and incorrect responses are acknowledged by the computer. This TRS-80 program comes on a 5¼-inch diskette and requires 48K memory. Price is $70.

Computer Action, 45 Paerdegat, 2nd Street, Brooklyn, NY 11236. Reader Service number 486.

BOOK REVIEWS

Microcomputer Analog Converter Software and Hardware Interfacing

Titus, Titus, Rony and Larsen
Howard W. Sams & Co., Inc.
Indianapolis, IN, 288 pp.

This book, part of the Blacksburg Continuing Education series, carries on the tradition of the "Bugbooks" published by E & L Instruments. It is a follow-up to *The 8080A Bugbook*, which you will need to have read to appreciate the information given here.

The book covers A/D and D/A interfacing, the key to using a microprocessor to control the external world. As in the earlier books, the authors continue the black box, or "bug," approach to hardware.

They pay little attention to external discrete components, and discuss only ICs. Rather than attempt to explore the internal workings of these units, the book studies their interactions.

The first two chapters cover the essentials of A/D and D/A converters and their interfacing, and are classics of programmed learning. They follow the general structure of the earlier Bugbooks, starting with an introduction and a list of objectives. The complexity of the software and hardware gradually increases; before you know it, you are easily handling variations on information you did not even know a few pages earlier.

My one complaint is that they use octal notations for their software. I don't care if octal is more logical and suited to the 8080 instructions; I have become indoctrinated with hex.

Chapters three, four and five discuss how these modules are used for the tasks a microcomputer would need to perform. The sixth chapter gives a short list of some commercially available units that incorporate A/D and D/A converters, and suggests how to interface them. This chapter is more useful to professional systems designers to whom cost efficiency, rather than absolute cost, is the main factor.

Unlike the earlier books, this one puts the experiments at the end. This detracts from the value of each chapter, since you lose the hands-on experience as you progress. It does, however, have its purpose; many of the experiments are built on the ones before, and building up the hardware and software from scratch for each experiment takes time.

The experiments follow the usual format, with the needed hardware listed at the end. The book assumes that you have a solderless breadboard and the necessary I/O from an 8080-type microcomputer.

Although the experiments were originally designed for E & L Instruments' MMD-1 microcomputer and the LR-35 Outboard, you can easily improvise. An oscilloscope would be convenient but you can use a VOM.

The experiments begin by having you interface a D/A converter to your computer. Then you make this D/A converter function as an A/D converter and use both to create input to and output from your computer. The authors describe the software, and you can use many of the complete units permanently. I doubt if you will ever find a situation that can't be solved by one of the examples.

The book ends with data sheets from several manufacturers of analog hardware.

This book would be a good course outline for both a theoretical and a laboratory course on microprocessors, and would be equally useful as a self-instruction book for the serious hobbyist interested in making his microcomputer do more than play games and function as a classy calculator. I recommend it highly.

Bruce Evans, M.D.
Pickering, Ontario

General Ledger: Accounting Programs for Small Computers

Louis D. Gray
Creative Computer Consultants, Inc.
$45

These nine programs, written in IBM BASIC, are only a literal translation of a manual accounting process.

So what, you say? Computers are great at repetitive processes. Accounting is a repetitive process. When humans perform the same operation over and over, they make errors.

But the best place to catch an error is immediately as it is made, and CCC's General Ledger doesn't provide this capability. Instead, you have to catch mistakes in the balance at the time of the trial balance. If the trial balance is not correct, then the user must go back and find the error.

This isn't too bad if you have only 100 transactions a month, but try it with 1000!

A good computerized accounting system will force the user to enter balancing debits for each credit at the time of entry. While CCC's process isn't bad, it could be better.

Unfortunately, General Ledger has other problems. CCC provides "complete" flowcharts for the programs that fail to define the process they were intended to explain. Nothing is clarified. How is the chart of accounts report generated? Is it done sequentially from what's on file, is it sorted and printed by ascending account number, or what?

In another example, the menu leads to an action code that leads back to the menu. The only way out of this loop that I can see is with something called "Term and Put on File," whatever that means.

In short, the flowcharts are mostly a waste of space.

A more annoying problem is that the chart of accounts is originally created from data statements rather than input statements. This is quite unusual, since the program asks for the company name via input statements. While the company name never changes, the chart of accounts might, so this is a bit suspicious. I never did like the IBM 5110, but a program like this might make me like it even less.

Are these all of the problems? I wish I could say yes, but listen on.

The edit programs allow you to do full editing or building of transaction files, but are more suitable to a Teletype, rather than a microcomputer, environment. Programs that do not clear the screen, present menus (there are no menus; you must remember all possible commands) or otherwise attempt to keep input/output carefully organized and simple are just not suitable for the business environment.

Yet another possible source of difficulty is that if you don't read and understand all of the material carefully, you might set up a chart of accounts that is not consistent with the balance sheet report or the income statement (i.e., the account numbers should be in the sequence in which they are to appear on the balance sheet and income statement, not just haphazard or randomly assigned).

Quite frankly, I wouldn't pay $10 for this book, let alone the $45 suggested retail price. The Osborne General Ledger is better documented, more thoughtfully laid out, and uses a video display forms technique for presenting and requesting information. Now that the Osborne programs are available in CBASIC 2, there is no reason to even consider CCC's offering.

Thom Hogan
Bloomington, IN

BASIC Software Library Vol. VI: A Complete Business System

R. W. Brown
Scientific Research Instruments
Key Biscayne, FL
$49.95

If you think a complete business system should include such basics as error correction, data modification, audit trails and interactive files, then save your money. This book offers none of these.

What the book does offer is full of errors. Of the five main programs, two do not run as listed.

The first program, ACBB, prints up bills, mailing labels, sales reports, accounts receivable reports, last purchase reports and account updates. But the program makes calls to the wrong data entry routines for information not stored in those files, and two lines created erroneous totals in the A/R and last sales reports. Such errors could not have happened if the author had actually run the programs as listed.

ACBI, the inventory program, does activity reports, minimum quantity search, inventory lists and inventory updates. But without several changes, this one won't run either.

The inventory depreciation program for fixed and depreciable assets is fairly short, and runs without any problems.

ACBL is the most lengthy and best-written of the five. Instead of using a menu, it automatically governs program flow and data acquisition. It has seven data files open at the same time, but you can rewrite the program without too much trouble to get around this.

The last program creates the initial data files. The book suggests that you substitute information pertaining to your company, since the programs do not allow you to ever correct, modify or delete any of the data once they are in the files. In fact, you don't even get a chance to review it.

All listings are printed in dot matrix, which makes them hard to read. For the price, the authors could have used a better printing method.

So what do you get for $49.95? It all depends on your needs. But the package does not include what the title says it does. After paying $49.95, I feel ripped off.

Greg Greene
North Vancouver, B.C.

OSI BASIC in ROM

Edward H. Carlson
3872 Raleigh Drive
Okemos, MI 48864
68 pp., $8.95

In our happy land of computing, there dwell several hungry, pernicious vendors of computer hardware. All day long, and sometimes far into the night, these monstrous moguls of madness sit in their castles and ponder how they might protect their computers. Battle tactics include:

- Misinforming the customer about how much support they will provide.
- Misleading the customer when he or she asks questions about the deep, dark secrets inside the product.
- Mislabeling products, just to cause confusion.
- Playing dumb.

Comes one Edward H. Carlson to the great, inviolate doors of OSI. He carries in his hands a message of truth. It is a book, a shabby, plain-covered, typewritten little volume of no outward note. But within, there is power.

As he raises his arms to wield this power, the arrows come singing from over the castle wall. But the hero is shielded by armor wrought from hours and hours of work.

As he unleashes his awesome might, hecklers are heard from the castle: " 'Tis confusing! And it costs too much!" But the people cry, "Nay!" and rush the crumbling doors.

"Oh, King, we like your equipment," the people say. "But we need his book! In times past you have said we didn't need to know what is written here, but that was a lie! The book explains how space is allocated, what values are supported and why little quirks develop. It does this for each of the BASIC statements. The appendix contains information on tape I/O, arrays, a BASIC trace, a memory map and a complete disassembly of the ROM with comments!"

What will His Excellency say? Will the people get the software and hardware support they truly need?

It seems, for some reason, that hardware manufacturers feel threatened when information of this type becomes available, yet history has shown that it actually helps sales.

OSI would do well to market Carlson's book themselves!

Dennis Thurlow
ISI staff
Peterborough, NH

LETTERS TO THE EDITOR

16-Bit Update

Martin Moore's article on 16-bit processors ("The 16-Bit Super Processors Are Here," August *Microcomputing*, p. 26) was good, and I hope you publish more articles like it; but it had many incorrect statements.

The discussion of memory space failed to mention that both the Intel 8086 and Zilog Z8000 have segmented memory spaces. The Motorola MC68000, on the other hand, can directly address the 16-megabyte memory map without incurring the additional delay required to set up segment registers. Note that while the Zilog MMU allows expansion of the address space to 48 megabytes through the use of the status lines, the function code outputs from the Motorola MC68000 may be decoded to recognize four 16-megabtye address spaces for user data, user program, supervisor data and supervisor program, thus conceivably allowing construction of an MC68000-based system capable of addressing 64 megabytes.

Motorola has been busy accepting orders and delivering parts to many customers. Parts are not being rationed. Motorola has proposed a new microprocessor bus, the Versabus, for IEEE acceptance. The Versabus with 32 data and 32 address lines would be usable with any of the 16-bit MPUs and will support future

32-bit MPUs. Motorola did not have any problems with buffer register design at all. The first mask set worked with the exception of the STOP instruction, which, when executed, halted the MPU until power was cycled off and on. This problem has been fixed for about six months now, and Motorola is delivering 4, 6 and 8 MHz MC68000s. The single clock signal is connected to a TTL-compatible input.

Martin also failed to note that the processor's eight data registers may also be used as address registers. The MC68000 instruction set is a brand new instruction set optimized for throughput and performance. There is not enough similarity between the MC68000 and the MC6800 to allow use of a translator, since the resulting code would be inefficient. The MC68000 is easier to program because it has only 59 instructions, as opposed to 72 for the MC6800.

In addition, unimplemented instructions trap to certain vector locations protecting system integrity, allowing the user to construct his own macro instructions and providing Motorola with the capability of adding additional instructions. All M6800 peripherals will work with the MC68000, thus providing a full set of design support chips.

Remember that 16-bit MPUs are very powerful, fast, typically easier to program than eight-bit processors, while allowing the user to access great amounts of memory.

Jack W. Browne, Jr.
Microprocessor Applications Engineer
Steve Sparks
Manager
Microprocessor Marketing and Systems Applications
Motorola, Inc.
Austin, TX

Rules of the Game

Don Lancaster has so well expressed my feelings in "Winning the Micro Game" (August *Microcomputing*, p. 36) that I had to take the time to say thanks. I am referring to the paragraphs starting with "If it's old line, stomp on it."

However, as a senior staff programmer with a large company, I have to work with the old-line equipment, systems, people and problems. I have been trying for three years to convince my company to use micros. I have not been successful.

The big question is, "How do you convince the old line to take a step away from the IBM truck and try out the micro sportster?"

Richard Goldner
Miami, FL

Any way you look at it, there is a real *mental* world. From pure theory all the way down into structured schemes, it is every bit as real as the physical one Don Lancaster is pointing to when he suggests taking "a 100-watt light bulb and shining light on the *real world*" ("Winning the Micro Game," August *Microcomputing*, p. 36).

Don't misunderstand. I know it's a figure of speech, and I don't criticize Mr. Lancaster's statement. He has taught me (through his publications) more real-world engineering than any university has. All I'm griping about is syntax; any good programmer knows that mental is every bit as real. As our language continues to take shape, full of new terms and usages, let's tell it like it is!

Dave Doody
Avalon, CA

A Good Diary

I wish to compliment you, and particularly Al Prentice, for the excellence of the article "File Sorting Program and Its Diary" (June and July *Microcomputing*).

I am a regular reader of your magazine and found this to be the best article I have read in *Microcomputing* and, for that matter, in any other computer magazine. The article was particularly informative for me, a recent newcomer to programming, since I could learn effective procedure. I hope you will have some similar articles in the future.

Harry G. Schaefer
Calgary, Alberta

With some commercial software requiring hours and hours to sort a couple of thousand records, this is an aspect of programming that needs more exploration . . . and articles. —Editors.

Software Pirates

As president of the Philadelphia Apple Club and partner in Progressive Software, I would like to express my views on piracy of copyrighted software. The official policy of my club is that no copyrighted software be traded between any members or any other club. It is the feeling of myself and my club that any piracy is counter-productive, since many authors work countless hours in developing these programs.

As for my company, we are sure we have lost thousands of dollars in sales because of piracy. Any piracy by anyone hurts us all, regardless of the type of micro we use.

If you have need for a program that you think will be of use to you in one way or another, buy it, since this gives other authors the incentive to produce high-quality material at competitive prices. We would all benefit.

Neil D. Lipson
Progressive Software
Plymouth Meeting, PA

Whenever I write a software review, I invariably receive letters from readers asking if I am willing to trade a copy of the program involved for a copy of something that they have. I always write back to them and make my position very clear: Selling, trading or giving away copies of copyrighted software is illegal and is the same as stealing money out of the pockets of the author and vendor of that software! Perhaps I feel very strongly about this because I earn a small income from software that I have created from time to time.

I also find it impossible to answer requests for information from readers who do not send a stamped self-addressed envelope. The cost is one reason, but the convenience of having an envelope already addressed is helpful in getting through a large stack of correspondence. I make this point not only for myself, but also for all authors, and because I want anyone who has written to me for advice and not received it to know why. I do answer all other correspondence.

Rod Hallen
Washington, DC

Name-Calling

I am in total agreement with your views on the phrase "personal computers" as expressed in your last dealer newsletter (*Microcomputing Industry Newsletter*, June 1980). However, I would carry it one step further—instead of a funeral for the phrase, let's have a cremation!

I was recently low bidder on a water-billing system for a local village. Then a representative from Infernal Big Mother (the only other bidder) went to the village board armed with some computer magazine. He showed Apple II ads using the obnoxious phrase and stated "do you want a personal computer or a *real* computer to do your water bills?" He also pointed out various game programs (naturally, omitting the ads offering business and professional programs) and convinced the board that what we had proposed would not work. As a result, the village now has a system that will process their water bills, and probably most of Chicago's water bills as well.

Let's call them microcomputers or desktop computers and rid ourselves of the quasi-computer image of "home" or "personal" computers.

E.C. Martin
President
Illinois Computer Mart, Inc.
Carbondale, IL

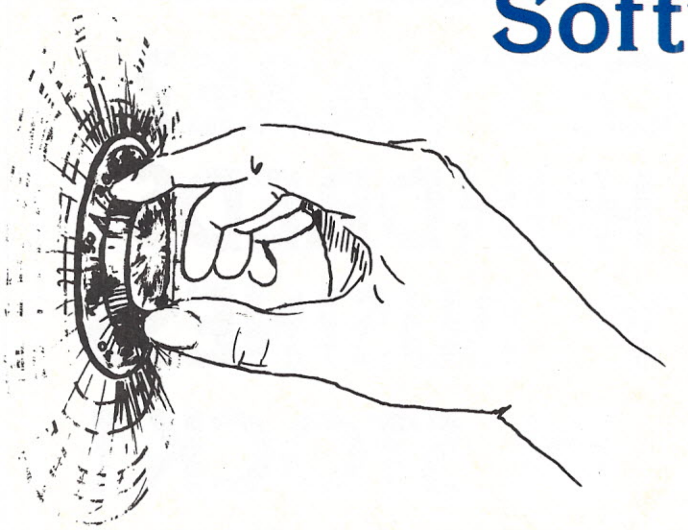
Let's call them microcomputers, because that's what they are. The use of other terms is not helpful.—Editors.

Law and Reorder

My wife and I are both hams, WB6IUN and WB6HJW, so it's appropriate that we stock your publications at our newsstand. Before moving to Oregon and starting our store just over a year ago, I was a police detective in Santa Maria, CA, and it was I who investigated and put a stop to the defunct DataSync Corp. that was written up in *Kilobaud Microcomputing*. Never thought back then that I'd someday be selling *Microcomputing*.

Ernie Kapphahn
Capitol News Center
Salem, OR

Software Security



With a clever password in your BASIC program, you can thwart unauthorized access —whether from industrial spies or nosy neighbors.

Walter K. McCahan
PO Box 3314
Shiremanstown, PA 17011

Now that minicomputers and microcomputers have entered the world of business to stay, it is time to consider the security aspects of these machines and their related systems and programs.

The first breach of security occurs when an unauthorized operator gains knowledge that he has no right or need to know. This could happen, for instance, during an unauthorized run of a program that contained personnel records. Or perhaps an unauthorized run of a budget program could reveal confidential market strategies.

The second, and more serious, unauthorized use of the computer results in direct monetary gain by the unauthorized user. An example of this is an operator who changes a payroll program to adjust his own rate of pay. Another example is an unauthorized person writing himself a check against the payroll or a check against accounts payable.

When setting rules to prevent this first kind of security violation, you should consider both the physical and software aspects of security; the second kind of violation is preventable mainly by security measures built into the software.

A Byte of Prevention

Controlling the physical security of the computer can be divided into four broad classifications:

1. Controlling the entrance to the room in which the computer is housed.
2. Protecting the medium upon which the program is stored.
3. Protecting the medium upon which the data is stored.
4. Controlling the forms upon which the output is printed.

Maintaining control over who is allowed to enter the room where the computer is kept is usually a difficult task when minis or micros are being used, since one of the desirable features of these systems is accessibility. If the proper office layout can be arranged, however, limiting access to the computer room can be an effective deterrent to unauthorized use.

Controlling the media upon which the programs and data are stored is also hardly ever the optimum solution to computer security. If this medium, which is most often either tape, disks or cards, is stored at a central point, only one person should have access to the container. Otherwise, every operator would have access to all of the media. On the other hand, if each operator keeps his own medium in a secure place, it will not be accessible to anyone else if that operator does not show up for work.

In large offices there is often a person appointed to keep all of this media under lock and key. This person is usually not otherwise associated with computer operations and has the job of handing out the media to each operator only on the basis of a predetermined access list.

If an unauthorized person is unable to gain access to the important stock forms used on the computer, it is difficult for him to print fraudulent documents. For example, if blank checks are not available, it is certainly more difficult for someone to print himself an unauthorized check.

The most effective, although not always easiest, security is built into the software. Since the preponderance of mini and microcomputer software is written in BASIC, the listings in this article are in BASIC, although the concepts remain the same in any language.

The underlying idea of building security into software is to make the program abort unless specific, prearranged information is input upon request. This can be done in

several ways, as the listings in this article will illustrate.

Listing 1 displays confidential payroll information and prints expense checks. It contains no security provisions.

```
10 '              **      EXPENSE REIMBURSEMENT DATA      **
20 DATA  22000,29000,32000
30 DATA 22000,29000,32000
40 DATA 5762,6868,6411
50 READ N1$(I),N2$(I),N3$(I)
60 READ A1,A2,A3
70 READ E1,E2,E3
80 CLS:INPUT"ENTER EMPLOYEES NAME";EN$
90 CLS:PRINT:PRINT:PRINT"NAME - ";EN$
100 IFEN$="JAMES SMITH" THEN PRINT"ANNUAL SALARY - ";A1
110 IFEN$="JAMES SMITH" PRINT"LAST YEARS EXPENSE ACCT -";E1
120 IFEN$="ROBERT GREEN" PRINT"ANNUAL SALARY - ";A2
130 IFEN$="ROBERT GREEN" PRINT"LAST YEARS EXPENSE ACCT -";E2
140 IFEN$="WILLIAM BLACK" PRINT"ANNUAL SALARY -";A3
150 IF EN$="WILLIAM BLACK" PRINT"LAST YEARS EXPENSE ACCT -";
    E3
160 PRINT:INPUT"ENTER AMOUNT OF CURRENT EXPENSE ACCOUNT TO B
    E REIMBURSED";C
170 GOSUB 3000
180  GOTO 80
3000 CLS:PRINT"THIS SUBROUTINE WOULD NORMALLY PRINT THE REIM
     BURSEMENT CHECK"
3010 PRINT:PRINT:INPUT"                    TO PROCEED PRESS ENTER
     ";X
3020 RETURN
```

Listing 1.

Adding a Simple Password

In order to protect the program from an unauthorized run, you can add a few lines of code near the start of the program to ask the operator to enter a password. These lines then compare the password to a preprogrammed password. If these passwords do not match, the program is aborted. To provide for this password in our example program, add the following lines to serve the functions indicated for each (see Listing 2):

12—Initializes the predetermined password.

14—Inputs password by operator.

16—Compares passwords.

4000—Aborts program.

```
10 '              **      EXPENSE REIMBURSEMENT DATA      **
12 P$="MKW4133"
14 CLS:INPUT"ENTER SECURITY PASSWORD";PW$
16 IF P$<>PW$GOTO 4000
20 DATA  22000,29000,32000
30 DATA 22000,29000,32000
40 DATA 5762,6868,6411
50 READ N1$(I),N2$(I),N3$(I)
60 READ A1,A2,A3
70 READ E1,E2,E3
80 CLS:INPUT"ENTER EMPLOYEES NAME";EN$
90 CLS:PRINT:PRINT:PRINT"NAME - ";EN$
100 IFEN$="JAMES SMITH" THEN PRINT"ANNUAL SALARY - ";A1
110 IFEN$="JAMES SMITH" PRINT"LAST YEARS EXPENSE ACCT -";E1
120 IFEN$="ROBERT GREEN" PRINT"ANNUAL SALARY - ";A2
130 IFEN$="ROBERT GREEN" PRINT"LAST YEARS EXPENSE ACCT -";E2
140 IFEN$="WILLIAM BLACK" PRINT"ANNUAL SALARY -";A3
150 IF EN$="WILLIAM BLACK" PRINT"LAST YEARS EXPENSE ACCT -";
    E3
160 PRINT:INPUT"ENTER AMOUNT OF CURRENT EXPENSE ACCOUNT TO B
    E REIMBURSED";C
170 GOSUB 3000
180  GOTO 80
3000 CLS:PRINT"THIS SUBROUTINE WOULD NORMALLY PRINT THE REIM
     BURSEMENT CHECK"
3010 PRINT:PRINT:INPUT"                    TO PROCEED PRESS ENTER
     ";X
3020 RETURN
4000 CLS:PRINT:PRINT:PRINT:PRINT:PRINT"              SECURITY
     VIOLATION - PROGRAM ABORTED"
4010 GOTO 4010
```

Listing 2.

In this example the password is a combination of the author's initials and post office box number—in reverse order. In developing a password, you should remember that it has to be simple enough for the authorized operator to remember, but complex enough to be obscure to an unauthorized operator. You should avoid using such obvious numbers as house numbers and telephone numbers. Birth dates, bank account numbers and a spouse's initials are more commonly used. Reversing the order of one or more elements is also a common practice.

Hiding the Password

In a situation where the operator is surrounded by employees or other curious onlookers, it is desirable not to have the password printed onto the screen as it is input by the operator. Change line 14 to:

CLS: Print "ENTER SECURITY PASSWORD": GOSUB5000

and add the subroutine starting at line 5000 (Listing 3) to the end of Listing 2.

Line 14 requests the input of the password and sends the program to the input subroutine. Lines 5000-5080 enter the password, which will not appear on the screen.

Using this method of matching passwords will prevent most casual operators from running the program, but the more knowledgeable operator will soon find that most systems will run BASIC programs from any point. The entire built-in software security could be defeated if the program were run starting with line 20.

Probably the best tactic to fill this security gap is to put a matching statement at the start of every program section. To accomplish this, add the following statement in lines 75, 95, 155, 165 and 3005, which will cause the program to abort if passwords are not matched:

75 IFP$<>PW$ORP$ = ""GOTO4000

If an authorized user is familiar with the programming aspects of BASIC as well as the operation of the computer, he will undoubtedly not be stymied by this software security. His next move would be to list the program and "pick out" the password by looking at line 12. To make this job ultimately more difficult, there are several methods of hiding the password.

The first method is by using variables, scattered throughout the program, that represent the letters and numbers that make up the password (see Listing 4).

To hide the password from an operator who is familiar with programming, an even more drastic measure must be taken: assign each variable with a coded character. This is accomplished by assigning the variable the ASCII equivalent of the letter or number to be assigned by the use of the CHR$ statement.

For example, consider line 162:

162 W1$ = "M":IFP$ = ""GOTO61

Since the ASCII equivalent of M is 77, change line 162 to:

162 W1$ = CHR$(77):IFP$ = ""GOTO61

Using this method, make the following changes to Listing 4:

```
5000 P1$=INKEY$:IFP1$=""GOTO5000
5010 P2$=INKEY$:IFP2$=""GOTO5010
5020 P3$=INKEY$:IFP3$=""GOTO5020
5030 P4$=INKEY$:IFP4$=""GOTO5030
5040 P5$=INKEY$:IFP5$=""GOTO5040
5050 P6$=INKEY$:IFP6$=""GOTO5050
5060 P7$=INKEY$:IFP7$=""GOTO5060
5070 PW$=P1$+P2$+P3$+P4$+P5$+P6$+P7$
5080 RETURN
```

Listing 3.

```
61 W2$ = CHR$(75):IFP$ = ""GOTO97
97 W3$ = CHR$(87):IFP$ = ""GOTO190
190W4$ = CHR$(52):W5$ = CHR$(49):IFP$ = ""GOTO35
35 W6$ = CHR$(51):IFP$ = ""GOTO13
```

The final revised program (Listing 5) would take even an advanced programmer considerable time to decode. The security methods in this article are designed to deter the casual and semi-sophisticated unauthorized operator; they will probably not stop a knowledgeable operator-programmer, only slow him down and make his work harder.

When developing an overall security system, remember that ease of operation is one of the best features that minis and micros have going for them, so any security system has to balance this ease of operation against ease of committing a fraudulent operation. ■

```
10 '            **      EXPENSE REIMBURSEMENT DATA      **
12 GOTO162
13 P$=W1$+W2$+W3$+W4$+W5$+W6$+W6$
14 CLS:PRINT"ENTER SECURITY PASSWORD":GOSUB5000
16 IF P$<>PW$GOTO 4000
20 DATA  22000,29000,32000
30 DATA 22000,29000,32000
35 W6$="3":IFP$=""GOTO13
40 DATA 5762,6868,6411
50 READ N1$(I),N2$(I),N3$(I)
60 READ A1,A2,A3
61 W2$="K":IFP$=""GOTO97
70 READ E1,E2,E3
75 IFP$<>PW$ORP$=""GOTO4000
80 CLS:INPUT"ENTER EMPLOYEES NAME";EN$
90 CLS:PRINT:PRINT:PRINT"NAME - ";EN$
95 IFP$<>PW$ORP$=""GOTO4000
97 W3$="W":IFP$=""GOTO190
100 IFEN$="JAMES SMITH" THEN PRINT"ANNUAL SALARY - ";A1
110 IFEN$="JAMES SMITH" PRINT"LAST YEARS EXPENSE ACCT -";E1
120 IFEN$="ROBERT GREEN" PRINT"ANNUAL SALARY - ";A2
130 IFEN$="ROBERT GREEN" PRINT"LAST YEARS EXPENSE ACCT -";E2
140 IFEN$="WILLIAM BLACK" PRINT"ANNUAL SALARY -";A3
150 IF EN$="WILLIAM BLACK" PRINT"LAST YEARS EXPENSE ACCT -";
    E3
155 IFP$<>PW$ORP$=""GOTO4000
160 PRINT:INPUT"ENTER AMOUNT OF CURRENT EXPENSE ACCOUNT TO B
    E REIMBURSED";C
162 W1$="M":IFP$=""GOTO61
165 IFP$<>PW$ORP$=""GOTO4000
170 GOSUB 3000
180  GOTO 80
190 W4$="4":W5$="1":IFP$=""GOTO35
3000 CLS:PRINT"THIS SUBROUTINE WOULD NORMALLY PRINT THE REIM
     BURSEMENT CHECK"
3005 IFP$<>PW$ORP$=""GOTO4000
3010 PRINT:PRINT:INPUT"                 TO PROCEED PRESS ENTER
     ";X
3020 RETURN
4000 CLS:PRINT:PRINT:PRINT:PRINT:PRINT"              SECURITY
     VIOLATION - PROGRAM ABORTED"
4010 GOTO 4010
5000 P1$=INKEY$:IFP1$=""GOTO5000
5010 P2$=INKEY$:IFP2$=""GOTO5010
5020 P3$=INKEY$:IFP3$=""GOTO5020
5030 P4$=INKEY$:IFP4$=""GOTO5030
5040 P5$=INKEY$:IFP5$=""GOTO5040
5050 P6$=INKEY$:IFP6$=""GOTO5050
5060 P7$=INKEY$:IFP7$=""GOTO5060
5070 PW$=P1$+P2$+P3$+P4$+P5$+P6$+P7$
5080 RETURN
```

Listing 4.

```
10 '            **      EXPENSE REIMBURSEMENT DATA      **
12 GOTO162
13 P$=W1$+W2$+W3$+W4$+W5$+W6$+W6$
14 CLS:PRINT"ENTER SECURITY PASSWORD":GOSUB5000
16 IF P$<>PW$GOTO 4000
20 DATA  22000,29000,32000
30 DATA 22000,29000,32000
35 W6$=CHR$(51):IFP$=""GOTO13
40 DATA 5762,6868,6411
50 READ N1$(I),N2$(I),N3$(I)
60 READ A1,A2,A3
61 W2$=CHR$(75):IFP$=""GOTO97
70 READ E1,E2,E3
75 IFP$<>PW$ORP$=""GOTO4000
80 CLS:INPUT"ENTER EMPLOYEES NAME";EN$
90 CLS:PRINT:PRINT:PRINT"NAME - ";EN$
95 IFP$<>PW$ORP$=""GOTO4000
97 W3$=CHR$(87):IFP$=""GOTO190
100 IFEN$="JAMES SMITH" THEN PRINT"ANNUAL SALARY - ";A1
110 IFEN$="JAMES SMITH" PRINT"LAST YEARS EXPENSE ACCT -";E1
120 IFEN$="ROBERT GREEN" PRINT"ANNUAL SALARY - ";A2
130 IFEN$="ROBERT GREEN" PRINT"LAST YEARS EXPENSE ACCT -";E2
140 IFEN$="WILLIAM BLACK" PRINT"ANNUAL SALARY -";A3
150 IF EN$="WILLIAM BLACK" PRINT"LAST YEARS EXPENSE ACCT -";
    E3
155 IFP$<>PW$ORP$=""GOTO4000
160 PRINT:INPUT"ENTER AMOUNT OF CURRENT EXPENSE ACCOUNT TO B
    E REIMBURSED";C
162 W1$=CHR$(77):IFP$=""GOTO61
165 IFP$<>PW$ORP$=""GOTO4000
170 GOSUB 3000
180  GOTO 80
190 W4$=CHR$(52):W5$=CHR$(49):IFP$=""GOTO35
3000 CLS:PRINT"THIS SUBROUTINE WOULD NORMALLY PRINT THE REIM
     BURSEMENT CHECK"
3005 IFP$<>PW$ORP$=""GOTO4000
3010 PRINT:PRINT:INPUT"                 TO PROCEED PRESS ENTER
     ";X
3020 RETURN
4000 CLS:PRINT:PRINT:PRINT:PRINT:PRINT"              SECURITY
     VIOLATION - PROGRAM ABORTED"
4010 GOTO 4010
5000 P1$=INKEY$:IFP1$=""GOTO5000
5010 P2$=INKEY$:IFP2$=""GOTO5010
5020 P3$=INKEY$:IFP3$=""GOTO5020
5030 P4$=INKEY$:IFP4$=""GOTO5030
5040 P5$=INKEY$:IFP5$=""GOTO5040
5050 P6$=INKEY$:IFP6$=""GOTO5050
5060 P7$=INKEY$:IFP7$=""GOTO5060
5070 PW$=P1$+P2$+P3$+P4$+P5$+P6$+P7$
5080 RETURN
```

Listing 5.

The first personal computer for under $200.

The Sinclair ZX80. A complete computer— only $199.95 *plus $5.00 shipping.*

Now, for just $199.95, you can get a complete, powerful, full-function computer, matching or surpassing other personal computers costing several times more.

It's the Sinclair ZX80, the computer that independent tests prove is faster than all previous personal computers. The computer that "Personal Computer World" gave 5 stars for 'excellent value.'

The ZX80 cuts away computer jargon and mystique. It takes you straight into BASIC, the most common, easy-to-use computer language.

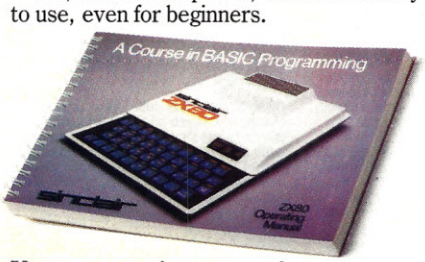
You simply take it out of the box, connect it to your TV, and turn it on. And if you want, you can use an ordinary cassette recorder to store programs. With the manual in your hand, you'll be running programs in an hour. Within a week, you'll be writing complex programs with confidence.

All for under $200.

Sophisticated design makes the ZX80 easy to learn, easy to use.

We've packed the conventional computer onto fewer, more powerful LSI chips—including the Z80A microprocessor, the faster version of the famous Z80. This makes the ZX80 the world's first truly portable computer (6½" x 8½" x 1½" and a mere 12 oz.). The ZX80 also features a touch sensitive, wipe-clean keyboard and a 32-character by 24-line display.

Yet, with all this power, the ZX80 is easy to use, even for beginners.



Your course in computing.

The ZX80 comes complete with its own 128-page guide to computing. The manual is perfect for both novice and expert. For every chapter of theory, there's a chapter of practice. So you learn by doing—not just by reading. It makes learning easy, exciting and enjoyable.

The ZX80's advanced design features.

Sinclair's 4K integer BASIC has performance features you'd expect only on much larger and more expensive computers. These include:

- Unique 'one touch' entry. Key words (RUN, PRINT, LIST, etc.) have their own single-key entry and are stored as a single character to reduce typing and save memory space.
- Automatic error detection. A cursor identifies errors immediately to prevent entering programs with faults.
- Powerful text editing facilities.
- Also programmable in machine code.
- Excellent string handling capability—up to 26 string variables of any length.
- Graphics, with 22 standard symbols.
- Built-in random number generator for games and simulations.

Sinclair's BASIC places no arbitrary restrictions on you—with many other flexible features, such as variable names of any length.

And the computer that can do so much for you now will do even more in the future. Options will include expansion of 1K user memory to 16K, a plug-in 8K floating-point BASIC chip, applications software, and other peripherals.

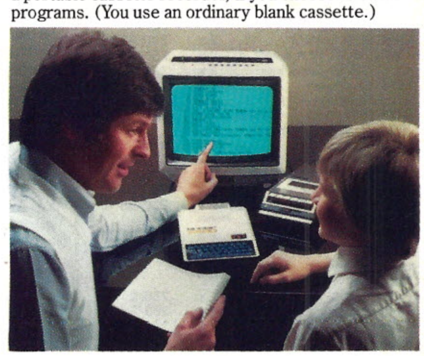
Order your ZX80 now!

The ZX80 is available only by mail from Sinclair, a leading manufacturer of consumer electronics worldwide. We've already sold tens of thousands of units in Europe, so demand will be great.

To order by mail, use the coupon below. But for fastest delivery, order by phone and charge to your Master Charge or VISA. The ZX80 is backed by a 30-day money-back guarantee, a 90-day limited warranty with a national service-by-mail facility, and extended service contracts are available for a minimal charge.

Price includes TV and cassette connectors, AC adaptor, and 128-page manual.

All you need to use your ZX80 is a standard TV (color or black and white). The ZX80 comes complete with connectors that easily hook up to the antenna terminals of your TV. Also included is a connector for a portable cassette recorder, if you choose to store programs. (You use an ordinary blank cassette.)



The ZX80 is a family learning aid. Children 10 and above will quickly understand the principles of computing—and have fun learning.

Phone orders only: (203) 265-9171. We'll refund the cost of your call.
Information: General and technical—(617) 367-1988, 367-1909, 367-1898, 367-2555.
Phones open Monday-Friday from 8 AM to 8 PM EST.

sinclair **Sinclair Research Ltd., 475 Main St., P.O. Box 3027, Wallingford, CT 06492.**

To: **Sinclair Research Ltd., 475 Main St., P.O. Box 3027, Wallingford, CT 06492.**
Please send me ________ ZX80 personal computer(s) at $199.95* each (US dollars), plus $5 shipping. (Your ZX80 may be tax deductible.)
I enclose a check/money order payable to Sinclair Research Ltd. for $________.

Name ________________

Address ________________

City ________ State ________ Zip ________

Occupation: ________________ Age: ________

Intended use of ZX80: ________________

Have you ever used a computer? ☐ Yes ☐ No. KM-11-O
Do you own another personal computer? ☐ Yes ☐ No. *For Conn. deliveries, add sales tax.

No. 17

Software with full support

Purchasing our software is just the beginning. We then back it up with professional support:

■ Subscription to "LIFELINES" for automatic notifications of revisions! ■ Update service for software and documentation! ■ Telephone Hotline! ■ Overseas software export service!

All Lifeboat programs require CP/M, unless otherwise stated.

Software with Manual / Manual Alone

☐ **CP/M* FLOPPY DISK OPERATING SYSTEM**—Digital Research's operating system configured for many popular micro-computers and disk systems:

New

System	Version	Price
Apple II* SoftCard* with Z80 Microsoft BASIC version 5 with high resolution graphics	2.x	349/25 ◇
North Star Single Density	2.x	170/25
North Star Double/Quad	2.x	170/25
Durango F-85	2.x	170/25
iCOM Micro-Disk 2411	1.4	145/25
iCOM 3712 for MITS 88-2SIO Console	1.4	170/25 *
iCOM 3712 for 3P+S/MITS SIO Rev non-zero console	1.4	170/25 *
iCOM 3812	1.4	170/25 *
Mits 3202/Altair 8800	1.4	145/25
Heath H8 + H17	1.4	145/25 Ⓜ
Heath H89	1.4	145/25 Ⓜ
Heath H89 by Magnolia	2.x	300/25 ◇
Ohio Scientific C3	2.x	200/25
Onyx C8001 Standard	2.x	250/25
Onyx C8001 Enhanced	2.x	330/25
TRS-80 Model I	1.4	145/25 Ⓜ
TRS-80 Model II	2.x	170/25
TRS-80 Model II + Corvus	2.x	250/25
Processor Technology Helios II	1.4	145/25
Intel MDS Single Density	2.x	170/25
Intel MDS Double Density	2.x	170/25
Micropolis Mod I	2.x	200/25 Ⓥ
Micropolis Mod II	2.x	200/25 Ⓥ
Mostek MDX STD Bus System	2.x	350/25 **

The following configurations are scheduled for release soon:

System	Version	Price
North Star Double/Quad + Corvus	2.x	250/25
Ohio Scientific C3-C	2.x	250/25
iCOM 3812	2.x	225/25 *
iCOM 4511/Pertec D3000	2.x	375/25 * +

*Software consists of the operating system, text editor, assembler, debugger and other utilities for file management and system maintenance. Complete set of Digital Research's documentation and additional implementation notes included. Systems marked * and ** include firmware on 2708 and 2716. Systems marked + include 5440 media charge. Systems marked Ⓜ require the special Ⓜ versions of software in this catalog. ◇ includes hardware addition to allow our standard versions of software to run under it.*

☐ **Z80 DEVELOPMENT PACKAGE**—Consists Ⓜ of: (1) disk file line editor, with global inter and Ⓩ intra-line facilities; (2) Z80 relocating assembler, Zilog/Mostek mnemonics, conditional assembly and cross reference table capabilities; (3) linking loader producing absolute Intel hex disk file .... **$95/$20**

☐ **ZDT**—Z80 Monitor Debugger to break and Ⓜ examine registers with standard Zilog/ Ⓩ Mostek mnemonic disassembly displays. $35 when ordered with Z80 Development Package .... **$50/$10**

AVOCET SYSTEMS

☐ **XASM-68**—Non-macro cross-assembler with nested conditionals and full range of pseudo operations. Assembles from standard Motorola MC6800 mnemonics to Intel hex ... **$200/$25**

☐ **XASM-65**—As XASM-68 for MOS Technology MCS-6500 series mnemonics ...... **$200/$25**

☐ **XASM-48**—As XASM-68 for Intel MCS-48 and UPI-41 families .... **$200/$25**

☐ **XASM-18**—As XASM-68 for RCA 1802 .... **$200/$25**

☐ **DISTEL**—Disk based disassembler to Intel 8080 or TDL/Xitan Z80 source code, listing and cross reference files, Intel or TDL/Xitan pseudo ops optional. Runs on 8080 .... **$65/$10**

Genuine CP/M for Apple II Available now!

Software with Manual / Manual Alone

☐ **DISILOG**—As DISTEL to Zilog/Mostek Ⓜ mnemonic files .... **$65/$10** Ⓩ

☐ **SMAL/80** Structured Macro Assembler Ⓧ Language—Package of powerful general purpose text macro processor and SMAL structured language compiler. SMAL is an assembler language with IF-THEN-ELSE, LOOP-REPEAT-WHILE, DO-END, BEGIN-END constructs .... **$75/$15**

PHOENIX SOFTWARE ASSOCIATES

☐ **PASM***—Z80 macro assembler, Intel/TDL Ⓩ mnemonics. Generates Intel hex format or relocatable code in either TDL Object Module format or PSA Relocatable Binary Module format. Supports text insertion, conditional branching within macros, recursive macro calls and parameter passing. .... **$129/$25**

☐ **EDIT**—Character oriented text file editor. Includes macro definition capabilities. Handles insertion, deletion, searching, block move, etc. for files of any length. Does not require a CRT. .... **$129/$25**

☐ **PLINK***—Two pass **disk-to-disk linkage editor/loader** which can produce re-entrant, ROMable code. Can link programs that are larger than available memory for execution targeted on another machine. Full library capabilities. Input can be PSA Relocatable Binary Module, TDL Object Module or **Microsoft REL files**. Output can be a COM file, Intel hex file, TDL Object Module or PSA Relocatable file. .... **$129/$25**

New

☐ **BUG* and μBUG***—Z80 interactive machine Ⓩ level debugging tools for program development. BUG has full symbolic trace and interactive assembly (mnemonics compatible with PASM). Dynamic breakpoints and conditional traps while tracing (even through ROM!). μBUG is a subset of BUG and is used in memory limited situations .... **$129/$25**

New

DIGITAL RESEARCH

☐ **MP/M**—Installed for single density MDS-800. Multi-processing derivative of the CP/M operating system. Manual includes CP/M2 documentation .... **$300/$50**

☐ **MAC**—8080 Macro assembler. Full Intel macro Ⓢ definitions. Pseudo Ops include RPC, IRP, REPT, TITLE, PAGE, and MACLIB. Produces absolute hex output plus symbol table file for use by SID and ZSID (see below) ..**$120/$15**

☐ **SID**—8080 Symbolic debugger. Full trace, Ⓢ pass count and breakpoint program testing. Has backtrace and histogram utilities. When used with MAC, provides full symbolic display of memory labels and equated values .**$105/$15**

☐ **ZSID**—Z80 Symbolic debugger with all fea- Ⓢ tures of SID .... **$130/$15** Ⓩ

☐ **TEX**—Text output formatter to create paginat- Ⓢ ed, page-numbered and justified copy. Output can be directed to printer or disk ...**$105/$15**

☐ **DESPOOL**—Utility program to permit simulta- Ⓢ neous printing from text files while executing other programs .... **$80/$10**

☐ **tiny C**—Interactive interpretive system for Ⓧ teaching structured programming techniques. Manual includes full source listings .**$105/$50**

☐ **BDS C COMPILER**—Supports structures, Ⓜ unions, 2 dimensional arrays, pointers, recur- Ⓛ sion and overlays. Features optimized code generator, variable sized buffers for file I/O, and capability to produce ROMable code. Includes macro package to enable user to produce linkable modules with MAC (see under Digital Research). Floating point functions, full run-time package and machine code library sources provided. Linker, library manager and textbook included. Compiler lacks initializers, statics, floats and longs. .... **$145/$25**

New Version

Software with Manual / Manual Alone

☐ **WHITESMITHS C COMPILER**—The ultimate Ⓛ in systems software tools. Produces faster Ⓧ code than a pseudo-code Pascal with more extensive facilities. Conforms to the full UNIX* Version 7 C language, described by Kernighan and Ritchie, and makes available over 75 functions for performing I/O, string manipulation and storage allocation. Linkable to Microsoft REL files. Requires 60K CP/M ..... **$630/$30**

MICROSOFT

☐ **BASIC-80**—Disk Extended BASIC, ANSI Ⓛ compatible with long variable names, Ⓜ WHILE/WEND, chaining, variable length file records. MBASIC version 4.51 also included on disk. .... **$325/$25**

☐ **BASIC COMPILER**—Language compatible Ⓛ with BASIC-80 and 3-10 times faster execution. Ⓜ Produces standard Microsoft relocatable binary output. Includes MACRO-80. Also linkable to FORTRAN-80 or COBOL-80 code modules .... **$350/$25**

☐ **FORTRAN-80**—ANSI 66 (except for COM- Ⓛ PLEX) plus many extensions. Includes relocat- Ⓜ able object compiler, linking loader, library with manager. Also includes MACRO-80 (see below) .... **$425/$25**

☐ **COBOL-80**—Level 1 ANSI '74 standard plus Ⓛ most of Level 2. Full sequential, relative, and Ⓜ indexed file support with variable file names. Powerful interactive, formatted screen handling with ACCEPT and DISPLAY verbs. Program segmentation for execution of programs larger than memory and CHAIN command with parameter passing. Full support of CP/M version 2 files. Includes MACRO-80 (see above), linking loader, and relocatable library manager. Requires 48K CP/M .... **$700/$25**

New

☐ **MACRO-80**—8080/Z80 Macro Assembler. Ⓛ Intel and Zilog mnemonics supported. Relocat- Ⓜ able linkable output. Loader, Library Manager and Cross Reference List utilities included .... **$149/$15**

☐ **XMACRO-86**—8086 cross assembler. All Ⓛ Macro and utility features of MACRO-80 package. Mnemonics slightly modified from Intel ASM86. Compatibility data sheet available .... **$275/$25**

☐ **EDIT-80**—Very fast random access text editor Ⓛ for text with or without line numbers. Global and intra-line commands supported. File compare utility included. .... **$89/$15**

☐ **PASCAL/M***—Compiles enhanced Standard Ⓧ Pascal to compressed efficient Pcode. Totally CP/M compatible. Random access files. Both 16 and 32-bit Integers. Runtime error recovery. Convenient STRINGs. OTHERWISE clause on CASE. Comprehensive manual (90 pp. indexed). SEGMENT provides overlay structure. INPORT, OUTPORT and untyped files for arbitrary I/O. Requires 56K CP/M. Specify 1) 8080 CP/M, 2) Z80 CP/M, or 3) Cromemco CDOS. .... **$175/$20**

☐ **PASCAL/Z**—Z80 native code PASCAL com- Ⓛ piler. Produces optimized, ROMable re-entrant Ⓩ code. All interfacing to CP/M is through the support library. The package includes compiler, relocating assembler and linker, and source for all library modules. Variant records, strings and direct I/O are supported. Requires 56K CP/M .... **$395/$25**

☐ **PASCAL/MT**—Subset of standard PASCAL. Ⓜ Generates ROMable 8080 machine code. Ⓧ Symbolic debugger included. Supports interrupt procedures, CP/M file I/O and assembly language interface. Real variables can be BCD, software floating point, or AMD 9511 hardware floating point. Includes strings enumerations and record data types. Manual explains BASIC-PASCAL conversion. Requires 32K .**$250/$30**

☐ **APL/V80**—Concise and powerful language for Ⓩ application software development. Complex programming problems are reduced to simple expressions in APL. Features include up to 27K active workspace, shared variables, arrays of up to 8 dimensions, disk workspace and copy object library. The system also supports auxiliary processors for interfacing I/O ports. Requires 48K CP/M and serial APL printing terminal or CRT .... **$500/$30**

☐ **ALGOL-60**—Powerful block-structured lan- Ⓩ guage compiler featuring economical run-time Ⓜ dynamic allocation of memory. Very compact (24K total RAM) system implementing almost all Algol 60 report features plus many powerful extensions including string handling direct disk address I/O etc. .... **$199/$20**

☐ **CBASIC-2** Disk Extended BASIC—Non- Ⓜ interactive BASIC with pseudo-code compiler and run-time interpreter. Supports full file control, chaining, integer and extended precision variables, etc. Versions of CRUN for CP/M versions 1.4 and 2.x included on disk. .**$120/$15**

MICRO FOCUS

☐ **STANDARD CIS COBOL**—ANSI '74 COBOL Ⓛ standard compiler fully validated by U.S. Navy tests to ANSI level 1. Supports many features to level 2 including dynamic loading of COBOL modules and a full ISAM file facility. Also, program segmentation, interactive debug and powerful interactive extensions to support protected and unprotected CRT screen formatting from COBOL programs used with any dumb terminal .... **$850/$50**

☐ **FORMS 2**—CRT screen editor. Output is Ⓛ COBOL data descriptions for copying into CIS COBOL programs. Automatically creates a query and update program of indexed files using CRT protected and unprotected screen formats. No programming experience needed. Output program directly compiled by STANDARD CIS COBOL .... **$200/$20**

Software with Manual / Manual Alone

☐ **NEVADA COBOL**—Subset of ANSI-74. Fea- Ⓜ tures fast compilation and execution with small object modules. Has extended arithmetic with 18 digit accuracy. Extended I/O includes random access files and sequential files of both fixed and variable length records, and interactive accept/display verbs. Good error messages and debugging facilities enhance program development. Requires a 32K CP/M system .... **$149/$25**

New

EIDOS SYSTEMS

☐ **KBASIC**—Microsoft Disk Extended BASIC Ⓛ version 4.51 integrated with KISS Multi-Keyed Index Sequential and Direct Access file management as 9 additional BASIC commands. KISS included as relocatable modules linkable to FORTRAN-80, COBOL-80, and BASIC COMPILER. Specify CP/M version 1.4 or 2.x when ordering. Requires 48K CP/M **$585/$45** To licensed users of Microsoft BASIC-80 (MBASIC) .... **$435/$45**

☐ **XYBASIC** Interactive Process Control BASIC—Full disk BASIC features plus unique commands to handle byte rotate and shift and to test and set bits. Available in several versions:

Integer ROM squared	**$350/$25**
Integer CP/M	**$350/$25**
Extended ROM squared	**$450/$25**
Extended CP/M	**$450/$25**
Extended Disk CP/M	**$550/$25**
Integer CP/M Run Time Compiler	**$350/$25**
Extended CP/M Run Time Compiler	**$450/$25**

☐ **RECLAIM**—A utility to validate media under CP/M. Program tests a diskette or hard disk surface for errors, reserving the imperfections in invisible files, and permitting continued usage of the remainder. Essential for any hard disk. Requires CP/M version 2. ...... **$80/$5**

☐ **BASIC UTILITY DISK**—Consists of: (1) Ⓜ CRUNCH-14— Compacting utility to reduce the size and increase the speed of programs in Microsoft BASIC 4.51, BASIC-80 and TRS-80 BASIC. (2) DPFUN—Double precision subroutines for computing nineteen transcendental functions including square root, natural log, log base 10, sine, arc sine, hyperbolic sine, hyperbolic arc sine, etc. Furnished in source on diskette and documentation .... **$50/$35**

☐ **STRING/80**—Character string handling plus routines for direct CP/M BDOS calls from FORTRAN and other compatible Microsoft languages. The utility library contains routines that enable programs to chain to a COM file, retrieve command line parameters and search file directories with full wild card facilities. Supplied as linkable modules in Microsoft format. **$95/$20**

☐ **STRING/80** source code available separately— **$295/NA**

☐ **THE STRING BIT**—FORTRAN character Ⓜ string handling. Routines to find, fill, pack, move, separate, concatenate and compare character strings. This package completely eliminates the problems associated with character string handling in FORTRAN. Supplied with source .... **$65/$15**

☐ **VSORT**—Versatile sort/merge system for fixed Ⓜ length records with fixed or variable length fields. VSORT can be used as a stand-alone package or loaded and called as a subroutine from CBASIC-2. When used as a subroutine, VSORT maximizes the use of buffer space by saving the TPA on disk and restoring it on completion of sorting. Records may be up to 255 bytes long with a maximum of 5 fields. Upper/lower case translation and numeric fields supported. .... **$175/$20**

☐ **CPM/374X**—Has full range of functions to create or re-name an IBM 3741 volume, display directory information and edit the data set contents. Provides full file transfer facilities between 3741 volume data sets and CP/M files .... **$195/$10**

Coming Soon

CPAids*

☐ **MASTER TAX**—Professional tax preparation Ⓛ program. Prepares schedules A, B, C, D, E, F, † G, R/RP, SE, TC, ES and forms 2106, 2119, 2210, 3468, 3903, 2441, 4625, 4726, 4797, 4972, 5695 and 6251. Printing can be on readily available, pre-printed continuous forms, on overlays, or on computer generated, IRS approved forms. Maintains client history files and is interactive with CPAids GENERAL LEDGER II (see below) .... **$995/$30**
Annual Update Fee **$350**

☐ **STANDARD TAX**—As above for schedules A, † B, C, D, E, G, R/RP, SE, TC and forms 2106 and 2441. Also, does not maintain client history files .... **$495/$30**
Annual Update Fee **$175**

☐ **GENERAL LEDGER II**—Designed for CPA's. † Stores complete 12 month detailed history of transactions. Generates financial statements, depreciation, loan amortizations, journals, trial balances, statements of changes in financial position, and compilation letters. Includes payroll system with automatic posting to general ledger. Prints payroll register, W2's and payroll checks. .... **$450/$30**

Lifeboat Associates, 1651 Third Avenue, N.Y., N.Y. 10028 **(212) 860-0300** Telex:220501

Neu in der Schweiz **Lifeboat Associates GmbH,** Aegeristr. 35, 6340 Baar **Telefon 042/31 2931**

Software with Manual / Manual Alone

New

☐ **T/MAKER**—Powerful new tool for preparing management reports with tabular data. Makes financial modeling projects easy. Do you want a weekly profitability report? Set up the table and compute. Just change the sales figures for next week and compute. You have a new report! T/MAKER includes a full screen editor for setting up tables which pages left, right, up and down. Compute includes standard arithmetic, percents, exponents, common transcendental functions, averages, maxima, minima, projections, etc. Requires 48K CP/M, CBASIC-2, CRT Terminal with addressable cursor positioning. . . . . **$275/$25**

☐ **BSTAM**—Utility to link one computer to another Ⓜ also equipped with BSTAM. Allows file transfers at full data speed (no conversion to hex), with CRC block control check for very reliable error detection and automatic retry. We use it! It's great! Full wildcard expansion to send *. COM, etc. 9600 baud with wire. 300 baud with phone connection. **Both** ends need one. Standard and Ⓜ versions can talk to one another. This software requires a knowledge of assembler language for installation. . . . . **$150/$10**

☐ **BSTMS**—Intelligent terminal program for Ⓜ CP/M systems. Permits communication between micros and mainframes. Sends character data files to remote computers under complete control. System can record character data sent from remote computer systems and data banks. Includes programs to EXPAND and COMPRESS binary files for transmission. This software requires a knowledge of assembler language for installation. . . . . **$200/$25**

☐ **WHATSIT?***—Interactive data-base system using associative tags to retrieve information by subject. Hashing and random access used for fast response. Requires CBASIC-2 . **$175/$25**

☐ **SELECTOR III-C2**—Data Base Processor to † create and maintain multi-key data bases. Ⓜ Prints formatted sorted reports with numerical summaries or mailing labels. Comes with sample applications, including Sales Activity, Inventory, Payables, Receivables, Check Register, and Client/Patient Appointments, etc. Requires CBASIC-2. Supplied in source . . . . . **$295/$20**

☐ **GLECTOR**—General Ledger option to SELECTOR III-C2. Interactive system provides for customized COA. Unique chart of transaction types insure proper double entry bookkeeping. Generates balance sheets, P&L statements and journals. Two year record allows for statement of changes in financial position report. Supplied in source. Requires SELECTOR III-C2. CBASIC-2 and 56K system. . . . . **$350/$25**

DMA

☐ **CBS**—Configurable Business System is a † comprehensive set of programs for defining custom data files and application systems without using a programming language such as BASIC, FORTRAN, etc. Multiple key fields for each data file are supported. Set-up program customizes system to user's CRT and printer. Provides fast and easy interactive data entry and retrieval **with transaction processing**. Report generator program does complex calculations with stored and derived data, record selection with multiple criteria, and custom formats. Sample inventory and mailing list systems included. **No support language required** . . . . **$395/$40**

MICROPRO

☐ **SUPER-SORT I**—Sort, merge, extract utility as Ⓛ absolute executable program or linkable module in Microsoft format. Sorts fixed or variable records with data in binary, BCD, Packed Decimal, EBCDIC, ASCII, floating & fixed point, exponential, field justified, etc. Even variable number of fields per record! . . . . . . **$225/$25**

☐ **SUPER-SORT II**—Above available as abso- Ⓛ lute program only . . . . **$175/$25**

☐ **SUPER-SORT III**—As II without SELECT/ Ⓛ EXCLUDE . . . . **$125/$25**

☐ **DATASTAR**—Professional forms control entry Ⓛ and display system for key-to-disk data capture. Menu driven with built-in learning aids. Input field verification by length, mask, attribute (i.e. upper case, lower case, numeric, auto-dup, etc.). Built-in arithmetic capabilities using keyed data, constant and derived values. Visual feedback for ease of forms design. Files compatible with CP/M-MP/M supported languages. Requires 32K CP/M . . . . **$350/$35**

☐ **WORD-STAR**—Menu driven visual word pro- Ⓛ cessing system for use with standard terminals. Text formatting performed on screen. Facilities for text paginate, page number, justify, center and underscore. User can print one document while simultaneously editing a second. Edit facilities include global search and replace, Read/Write to other text files, block move, etc. Requires CRT terminal with addressable cursor positioning . . . . **$445/$40**

☐ **WORD-STAR-MAIL-MERGE**—As above with Ⓛ option for production mailing of personalized documents with mail lists from DATASTAR or NAD . . . . **$575/$40**

☐ **WORD-MASTER** Text Editor—In one mode Ⓛ has superset of CP/M's ED commands including global searching and replacing, forwards and backwards in file in video mode, provides full screen editor for users with serial addressable-cursor terminal . . . . **$145/$25**

Software with Manual / Manual Alone

New

☐ **MAGIC WAND***—Word processing system with simple, easy to use full screen text editor and powerful print processor. Editor has all standard editing functions including text insert and delete, global search and replace, block move and library files for boiler plate text. Print processor formatting commands include automatic margins, pagination, headings & footings, centered and justified text. Also prints with true proportional spacing, merges with data files for automatic form letters, and performs run-time conditional testing for varied output. Requires 32K CP/M and CRT terminal with addressable cursor. . . . . **$395/$40**

☐ **TEXTWRITER III**—Text formatter to justify and Ⓜ paginate letters and other documents. Special features include insertion of text during execution from other disk files or console, permitting recipe documents to be created from linked fragments on other files. Has facilities for sorted index, table of contents and footnote insertions. Ideal for contracts, manuals, etc. Now compatible with Electric Pencil* and Word-Star prepared files . . . . **$125/$20**

New

☐ **DATEBOOK**—Program to manage time just like an office appointment book but using the speed and memory of a computer. Keeps track of three appointment schedules (three dental chairs, three attorneys, etc.) at once. Appointments consist of name, reason for the appointment, the date and time, and the length of the appointment. System can be quickly customized for the individual user. Many helpful features for making, changing, finding, and reporting appointments. Requires 48K CP/M and 180K bytes diskette storage. Not available for Apple CP/M. **$295/$25**

New lower prices for application Software

PEACHTREE SOFTWARE*

☐ General accounting software for small busi- Ⓛ nesses. Each product can be used alone or with † automatic posting to the general ledger. Supplied in source for Microsoft BASIC 4.51.

GENERAL LEDGER	$530/$40
ACCOUNTS PAYABLE	$530/$40
ACCOUNTS RECEIVABLE	$530/$40
PAYROLL	$530/$40
INVENTORY	$660/$40

☐ Other application products supplied in source Ⓛ for Microsoft BASIC 4.51.

† MAILING ADDRESS	$530/$40
PROPERTY MANAGEMENT	$925/$40

GRAHAM-DORIAN SOFTWARE SYSTEMS

☐ Comprehensive accounting software written in Ⓛ CBASIC-2 and supplied in source code. Each Ⓜ software package can be used as a stand- † alone system or integrated with the General Ledger for automatic posting to ledger accounts. Requires CBASIC-2.

GENERAL LEDGER	$805/$40
ACCOUNTS PAYABLE	$805/$40
ACCOUNTS RECEIVABLE	$805/$40
INVENTORY SYSTEM	$555/$40
JOB COSTING	$805/$40
APARTMENT MANAGEMENT	$805/$40
CASH REGISTER	$805/$40

☐ **POSTMASTER**—A comprehensive package Ⓜ for mail list maintenance that is completely menu driven. Features include keyed record extraction and label production. A form letter program is included which provides neat letters on single sheet or continuous forms. Includes NAD file translator. Requires CBASIC-2. . . . . **$150/$20**

STRUCTURED SYSTEMS GROUP

☐ Complete interactive accounting software for † business. Each product can be used standalone or with automatic posting to the general ledger. Each product is thoroughly tested and very well documented. Each product requires CBASIC-2.

GENERAL LEDGER	$820/$40
ACCOUNTS RECEIVABLE	$820/$40
ACCOUNTS PAYABLE	$820/$40
PAYROLL	$820/$40
INVENTORY CONTROL	$820/$40

LIFELINES

NEWSLETTER FROM LIFEBOAT

LIFELINES is the first step in software support for the serious microcomputer user. Each issue reports new revisions together with information on the purpose for each such release, be it for correction of "bugs" or the addition of features and facilities.

Feature Articles ! New Software ! Product Comparisons ! Info on CP/M Users Group !

SUBSCRIPTION INFORMATION:

$18 for twelve issues: U.S., Canada, and Mexico.

$40 for twelve issues: all other countries.

$2.50 for each back issue: U.S., Canada, and Mexico.

$3.60 for each back issue: all other countries. Send Check to LIFELINES, 1651 Third Avenue, New York, N.Y. 10028 or use your VISA or MASTERCARD—call (212) 722-1700

Software with Manual / Manual Alone

☐ **ANALYST**—Customized data entry and report- † ing system. User specifies up to 75 data items per record. Interactive data entry, retrieval, and update facility makes information management easy. Sophisticated report generator provides customized reports using selected records with multiple level breakpoints for summarization. Requires a disk sort utility such as QSORT, SUPER-SORT or VSORT and CBASIC-2 . . . . **$250/$15**

☐ **LETTERIGHT**—Program to create, edit and type letters or other documents. Has facilities to enter, display, delete and move text, with good video screen presentation. Designed to integrate with NAD for form letter mailings. Requires CBASIC-2 . . . . **$200/$25**

☐ **NAD** Name and Address selection system—Interactive mail list creation and maintenance program with output as full reports with reference data or restricted information for mail labels. Transfer system for extraction and transfer of selected records to create new files. Requires CBASIC-2 . . . . **$100/$20**

☐ **QSORT**—Fast sort/merge program for files with fixed record length, variable field length information. Up to five ascending or descending keys. Full back-up of input files created . . . . **$100/$20**

★ ★ ★ ★ ★ ★ ★

CONDIMENTS

★ ★ ★ ★ ★ ★ ★

☐ **HEAD CLEANING DISKETTE**—Cleans the drive Read/Write head in 30 seconds. Diskette absorbs loose oxide particles, fingerprints, and other foreign particles that might hinder the performance of the drive head. Lasts at least 3 months with daily use. Specify 5" or 8".
Single sided . . . . **$20 each/$55 for 3**
Double sided . . . . **$25 each/$65 for 3**

☐ **DC 300 Data Cartridges** Specify 450′XL or 300′ certified. Pack of 5. . . . . **$100**

☐ **FLIPPY DISK KIT**—Template and instructions to modify single sided 5¼" diskettes for use of second side in single sided drives **$12.50**

☐ **FLOPPY SAVER**—Protection for center holes for 5" and 8" floppy disks. Only 1 needed per diskette. Kit contains centering post, pressure tool and tough 7 mil mylar reinforcing rings for 25 diskettes.

5", Kit	$14.95
5", Rings only	$7.95
8", Kit	$16.95
8", Rings only	$8.95

☐ **PASCAL USER MANUAL AND REPORT**—By Jensen and Wirth. The standard textbook on the language. Recommended for use by Pascal/Z, Pascal/M and Pascal/MT users **$12**

☐ **THE C PROGRAMMING LANGUAGE**—By Kernighan and Ritchie. The standard textbook on the language. Recommended for use by BDS C, tiny C, and Whitesmiths C users . . **$12**

☐ **STRUCTURED MICROPROCESSOR PROGRAMMING**—By the authors of SMAL/80. Covers structured programming, the 8080/8085 instruction set and the SMAL/80 language **$20**

☐ **ACCOUNTS PAYABLE & ACCOUNTS RECEIVABLE—CBASIC**—By Osborne/McGraw-Hill **$20**

☐ **GENERAL LEDGER—CBASIC**—By Osborne/McGraw-Hill **$20**

☐ **PAYROLL WITH COST ACCOUNTING—CBASIC**—by Osborne/McGraw-Hill **$20**

★ ★ ★ ★ ★ ★ ★

Hearty Appetite.

★ ★ ★ ★ ★ ★ ★ ★

*CP/M and MP/M are trademarks of Digital Research.
Z80 is a trademark of Zilog, Inc.
UNIX is a trademark of Bell Laboratories.
WHATSIT? is a trademark of Computer Headware.
Electric Pencil is a trademark of Michael Shrayer Software.
TRS-80 is a trademark of Tandy Corp.
Pascal/M is a trademark of Sorcim.
SoftCard is a trademark of Microsoft.
Apple is a trademark of Apple Computer.
PASM, PLINK, BUG and μBUG are trademarks of Phoenix Software Associates Ltd.
CPAids is a trademark of Computer Tax Service, Inc.
MAGIC WAND is a trademark of Small Business Application, Inc.
Peachtree Software is a trademark of Retail Sciences, Inc.

†Recommended system configuration consists of 48K CP/M, 2 full size disk drives, 24 x 80 CRT and 132 column printer.

Ⓜ Modified version available for use with CP/M as implemented on Heath and TRS-80 Model I computers.

Ⓛ User license agreement for this product must be signed and returned to Lifeboat Associates before shipment may be made.

Ⓘ This product Includes/eXcludes the language Ⓧ manual recommended in Condiments.

Ⓢ Serial number of CP/M system must be supplied with orders.

Ⓩ Requires Z80 CPU.

Ordering Information

MEDIA FORMAT ORDERING CODES
When ordering, please specify format code.

LIFEBOAT ASSOCIATES MEDIA FORMATS LIST. Diskette, cartridge disk and cartridge tape format codes to be specified when ordering software for listed computer or disk systems. All software products have specific requirements in terms of hardware or software support, such as MPU type, memory size, support operating system or language.

Computer system	Format Code
Altair 8800 Disk	See MITS 3200
Altos	A1*
Apple + SoftCard 13 Sector	RG
Apple + SoftCard 16 Sector	RR
AVL Eagle	RB
BASF System 7100	RD
Blackhawk Single Density	Q3
Blackhawk Micropolis Mod II	Q2
CDS Versatile 3B	Q1
CDS Versatile 4	Q2
COMPAL-80	Q2
Cromemco System 3	A1*
Cromemco Z2D	R6
CSSN BACKUP (tape)	T1 #
Delta	A1*
Digi-Log Microterm II	RD
Digital Microsystems	A1*
Discus	See Morrow Discus
Durango F-85	RL
Dynabyte DB8/2	R1
Dynabyte DB8/4	A1*
Exidy Sorcerer + Lifeboat CP/M	Q2
Exidy Sorcerer + Exidy CP/M	Q4
Heath H8 + H17/H27	P4
Heath H89 + Lifeboat CP/M	P4
Heath H89 + Magnolia CP/M	P7
Helios II	See Processor Technology
Horizon	See North Star
iCOM 2411 Micro Floppy	R3
iCOM 3712	A1
iCOM 3812	A1*

Computer system	Format Code
iCOM 4511 5440 Cartridge CP/M 1.4	D1 #
iCOM 4511 5440 Cartridge CP/M 2.2	D2 #
IMS 5000	RA
IMS 8000	A1*
IMSAI VDP-40	R4**
IMSAI VDP-42	R4**
IMSAI VDP-44	R5**
IMSAI VDP-80	A1**
Intecolor	See ISC Intecolor
Intel MDS Single Density	A2
Intel MDS Double Density	A5
Intertec SuperBrain DOS 0.1	R7
Intertec SuperBrain DOS 0.5-2.X	RJ
Intertec SuperBrain DOS 3.X	RK
ISC Intecolor 8063/8360/8963	A1
Kontron PSI-80	RF
Meca 5¼"	P6
Micromation (Except TRS-80 below)	A1*
Micropolis Mod I	Q1
Micropolis Mod II	Q2
MITS 3200/3202	B1
Morrow Discus	A1*
Mostek	A1
MSD 5¼"	RC
North Star Single Density	P1
North Star Double/Quad	P2
Nylac Single Density	Q3
Nylac Micropolis Mod. II	Q2
Ohio Scientific C3	A3
Onyx C8001	T2#
Pertec PCC 2000	A1*
Processor Technology Helios II	B2
Quay 500	RO
Quay 520	RP
RAIR Single Density	R9

Computer system	Format Code
RAIR Double Density	RE
Research Machines 8"	A1
Research Machines 5¼"	RH
REX	Q3
Sanco 7000 5¼"	RQ
SD Systems 8"	A1*
SD Systems 5¼"	R3
Sorcerer	See Exidy Sorcerer
Spacebyte	A1
SuperBrain	See Intertec
Tarbell	A1*
TEI 5¼"	R3
TEI 8"	A1*
Thinkertoys	See Morrow Discus
TRS-80 Model I 5¼"	R2
TRS-80 Model I + FEC Freedom	RN
TRS-80 Model I + Micromation	A4*
TRS-80 Model I + Omikron 5¼"	RM
TRS-80 Model I + Omikron 8"	A1
TRS-80 Model I + Shuffleboard 8"	A1
TRS-80 Model II	A1*
VDP-40/42/44/80	See IMSAI
Vector Graphic	Q2
Vector MZ	Q2
Versatile	See CDS Versatile
Vista V80 5¼" Single Density	P5
Vista V200 5¼" Double Density	P6
Zenith Z89 + Lifeboat CP/M	P4
Zenith Z89 + Magnolia CP/M	P7

Prices reflect distribution on 8" single density diskettes. If a format is requested which requires additional diskettes, a surcharge of $8. per additional diskette will be added.

Prices F.O.B. New York. Shipping, handling and C.O.D. charges extra.

Manual cost applicable against price of subsequent software purchase.

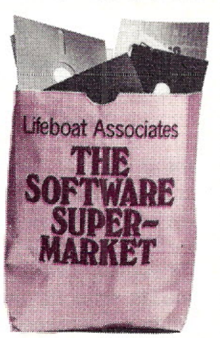
The sale of each proprietary software package conveys a license for use on one system only.

* Single-Side Single-Density disks are supplied for use with Double-Density and Double-Side 8" soft sector format systems.

** IMSAI formats are single density with directory offset of zero.

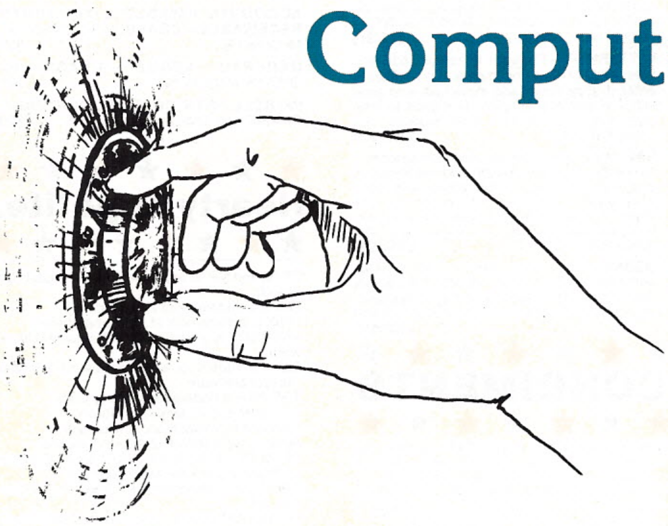
\# A media surcharge of $25 for orders on tape formats T1 and T2 and of $100 for orders on disk formats D1 and D2 will be added.

The list of available formats is subject to change without notice. In case of uncertainty, call to confirm the format code for any particular equipment.



Lifeboat Associates
1651 Third Avenue, N.Y., N.Y. 10028
(212) 860-0300

Richard R. Parry
38 W. 255 Deerpath Road
Batavia, IL 60510



Computerized Security And Status System

This dedicated 6800-based system makes your home or business intruder-proof.

```
15:43:23  08/14/80  TEST
16:21:40  08/14/80  TELEPHONE
16:21:46  08/14/80  TELEPHONE
16:21:52  08/14/80  TELEPHONE
16:21:58  08/14/80  TELEPHONE
17:33:10  08/14/80  FURNACE ON
17:41:22  08/14/80  FURNACE ELASPED TIME 08:12
18:02:50  08/14/80  FRONT DOORBELL
18:02:51  08/14/80  FRONT DOORBELL
18:04:02  08/14/80  BACK DOORBELL
18:04:08  08/14/80  BACK DOORBELL
18:06:12  08/14/80  BASEMENT WINDOW BROKEN
18:06:12  08/14/80  SIREN ON
18:08:42  08/14/80  SIREN OFF
18:22:04  08/14/80  MOTION DETECTED
18:22:04  08/14/80  SIREN ON
18:24:34  08/14/80  SIREN OFF
18:37:02  08/14/80  FURNACE ON
18:44:28  08/14/80  FURNACE ELAPSED TIME 07:26
```

Sample Run 1. Sample output of the system. The first event shows a test which assures the user the system is operating. The telephone then rings four times, followed by the cycling of the furnace. A hypothetical burglar rings the front and back doorbells, decides no one is home and proceeds to break the basement window to gain access. The siren is actuated for 2.5 minutes and scares off the would-be burglar. Minutes later he tries again. However, this time the motion detector senses the burglar's presence and sounds the alarm again.

It will never happen to me" was my reaction whenever I thought of the possibility of being burglarized. But when I came home one day to find air occupying the space where my stereo once stood, my opinion quickly changed.

I now have a computer-controlled home security and status system to help prevent it from happening again. The system monitors incoming phone calls, the front and back doorbells and the basement windows. I've also included a motion detector and a 110 decibel siren.

This system barely begins to explore the possibilities. For instance, you can use it to control your lights or call the police. With enough hardware, you can monitor every window and door in the house. Or you can hook the computer up to other monitoring devices such as light beams or smoke detectors.

About the System

You'll need a dedicated computer. You don't want to have to shut the system down every time you wish to play Star Trek. However, after making a major expenditure on a computer system, you're not likely to be able to finance another complete system. The solution is to get an evaluation kit, which was designed to introduce the neophyte to the capability and characteristics of a particular microprocessor.

Perhaps the most popular evaluation kit is the KIM-1. It was developed to acquaint a potential user with MOS Technology's 6502 microprocessor.

Motorola offers the MEK6800D1 and the MEK6800D2 evaluation kits for the 6800. Al-

though the program described in this article was designed for a noncommercial single board computer, the MEK6800D2 is very similar. An additional PIA must be added, and the addresses of the PIAs must be altered. Neither of these requires major change.

You'll also need a dedicated printer. A used five-level-code machine, such as a model 15, goes for about $60. The computer automatically converts characters from ASCII to five-level-code before printing. If you have an ASCII printer, you can modify the program to skip the code conversion routine.

An uninterruptible power supply is almost a must. A momentary power outage five minutes after you left for a two-week vacation would render the system useless. Unfortunately, a backup system may be expensive.

But, there is another way to ensure the integrity of the system. You can place the program in ROM in a computer that jumps to the ROM program when it is powered up. Although the time and date, which is in RAM, will be lost, the critical features of the system will still function.

The program is less than 1K and therefore fits nicely in a 2708 EPROM integrated circuit. The MEK6800D2 evaluation kit and most other computers accept this popular chip.

Sample Run 1 shows all of the system's features. The first event shows that the system is operational. A test switch causes the system to respond as shown.

Note that the time and date are appended to every event. The program features a real-time 24-hour clock and date routine. The date routine automatically updates the day, month and year. Only during a leap year will you need to correct it.

Sample 1 shows how incoming telephone calls are monitored. While the system will not tell you who called, knowing that you were called is often useful. Here is one area where the system can be embellished. For example, the system can control a tape recorder to record messages.

The front and back doorbells are also monitored to indicate visitors. Each time they are actuated, the event is noted with the time and date as shown.

Besides these status features of the system, two additional features are included for home security—the status of basement windows and a motion detector. Basement windows are perhaps the easiest way to gain access. Fortunately, they are also easy to hook up. When a window is broken, the event is indicated as shown in the sample.

In addition, the siren is actuated for 2½ minutes, enough to deter the average burglar, especially if it is loud (over 110 dB).

The motion detector is placed in a major traffic path or room to virtually cover the entire home. This input to the system would be deactivated when the home is inhabited. Like breaking a window, a tripped condition detected by the motion detector triggers the siren.

The Software

As shown in Listing 1, the program begins at $460. However, before actually starting the program, you must initialize the real-time clock variables: HR, MIN, SEC, MON, DAY and YR. They are located at the beginning of memory, locations 0 through 5, respectively. The clock must be initialized by storing in these locations the time and date in binary form. For example, $11, $21, $00, $07, $17, $50 represents 17:33:00 7/23/80.

When the real-time clock is initialized, you begin the program at $460. Lines 48 through 81 contain the initialization portion of the program. This routine sets the interrupt mask to inhibit interrupts until the initialization is complete.

In addition, the stack is set and the interrupt vector address at $A000 and $A001 is set to vector all interrupts to the start of the program shown in line 87. When the 6800 receives a 60 Hz interrupt via the PIA (peripheral interface adapter), it jumps to the location specified in locations $FFF8 and $FFF9. In most, if not all, 6800 systems, these addresses point to another address, namely $A000 and $A001. In line 54, $04A9 is stored at these locations, thereby causing control to pass to $04A9 each time an interrupt occurs. Due to the 60 Hz timebase, interrupts occur every 16.67 milliseconds.

The initialization routine next clears key variables and configures the PIAs. The program must also be informed of the location of the ASCII-to-five-level-code conversion table. The table is shown in lines 42 through 47. The table must be located on a 256-byte boundary. The high-order byte of the memory location must then be stored in variable ABAPNT. Listing 1 shows the table starting at $0400. Therefore, lines 79 and 80 cause ABAPNT to be initialized to $04. Similarly, if the table were placed at $FA00, then ABAPNT would be initialized to $FA. Remember that the program may be placed anywhere in memory, even in ROM, but the conversion table must always start on a 256-byte boundary.

The next major portion of the program is shown in lines 82 through 129. Each time an interrupt occurs, the computer is directed here, where it initiates three major functions. The siren is energized if the siren flag (SIRFLG) is set, and de-energized if the siren has been on for 2½ minutes. If there is no request for the siren, this portion of the routine is skipped completely.

Next, the real-time clock routine is updated. It is a separate subroutine starting at line 358. Finally, the status inputs to the system are updated (e.g., front doorbell, furnace). The status of the inputs is found by reading the A port of PIA0.

When the computer reads the PIA, each input is represented by a bit. While this is a compact way of storing the data, it makes examination of each bit cumbersome. Therefore, a short routine (lines 113 to 129) examines each of the eight bits and stores them in eight bytes.

The memory map shown in Fig. 1 clearly indicates how the input data is stored. For example, location $40 contains the most recent status on PIA input line PA0, which, in our case, is the TEST switch. The old data is the status that the system acquired 16.67 milliseconds earlier during the previous interrupt.

The next eight routines, labeled Test Switch, Telephone, Front Doorbell, Basement Window, Movement Detector, Furnace, Spare Input and Back Doorbell, all compare the old and new data. If a difference is detected, a call to a separate routine

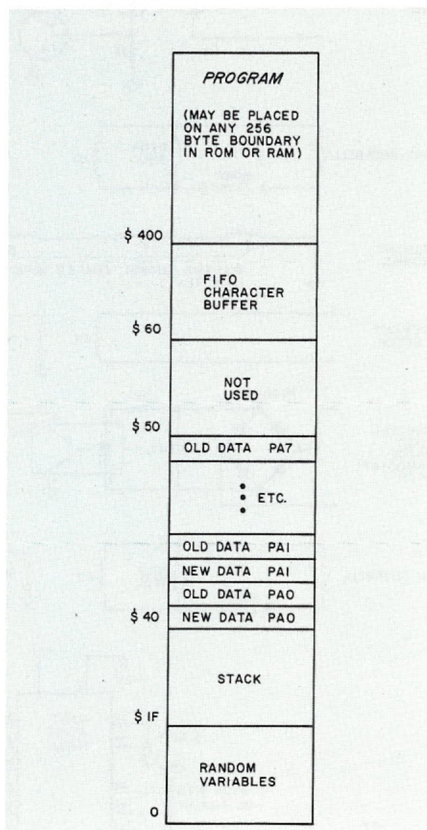


Fig. 1. The memory map indicates the location of the program and key variables. Note that the program can be placed in RAM or ROM, which makes it well suited to single board computers with limited RAM. Note also the addresses of the input data. For example, the new and old telephone statuses are located at $42 and $43, respectively. The time difference between new and old data is 16.67 milliseconds.

to store the event in the FIFO character buffer with the time and date is initiated.

The eight routines are similar. The basement window and motion detector routines perform the additional function of setting the siren flag to indicate there is a need to actuate the siren. The furnace routine sets a flag to indicate there is a need to append the elapsed time to the event.

A subroutine called Store, which begins on line 273, is used to store the text with the time and date appended. To convert the time and date, which is stored in binary form to ASCII characters, another subroutine called STOTIM is called. This subroutine converts a binary number to two printable ASCII characters. For example, binary 4 (100) is converted to $30 and $34, which are printable as the decimal number 04.

The next major routine starts on line 230. This routine determines if there is an event to print by examining the variable KBLOCK. If KBLOCK is nonzero, the teleprinter motor is energized and given three seconds to come up to speed before the event is actually printed. The subroutine ASCBAU is called to convert an ASCII character to five-level before the character is printed. If there are more than ten events to print, no more events are accepted by the system. To accept more events would cause the system to overflow a 500 byte character buffer. If more memory is available, then MAXEVT (see line 10) may be changed accordingly.

The last portion of the program to be discussed is the event text, which begins on line 462. When an event is detected, the index register points to one of these text messages. The text is printed until an asterisk (*) is detected. In other words, the asterisk acts as a delimiter. The last line of the program is the end of line (EOL) sequence that is appended to each event. It consists of two line feeds and two carriage returns.

About the Hardware

Fig. 2 shows a schematic of the hardware necessary for compatibility with the software.

The hardware consists of three major divisions. The system's inputs are connected to the A port of PIA0. A printer interface is connected to the UART via the B port of PIA0. And the B port of PIA1 is used for miscellaneous signals such as controlling the teleprinter, siren and detecting interrupts.

The input section consists of seven inputs; an eighth input (PA6) is not used and may be connected to any additional input you desire. This point is clearly marked in the listing on line 211. Additional code could be added here, very much like the other seven, as well as appropriate text to

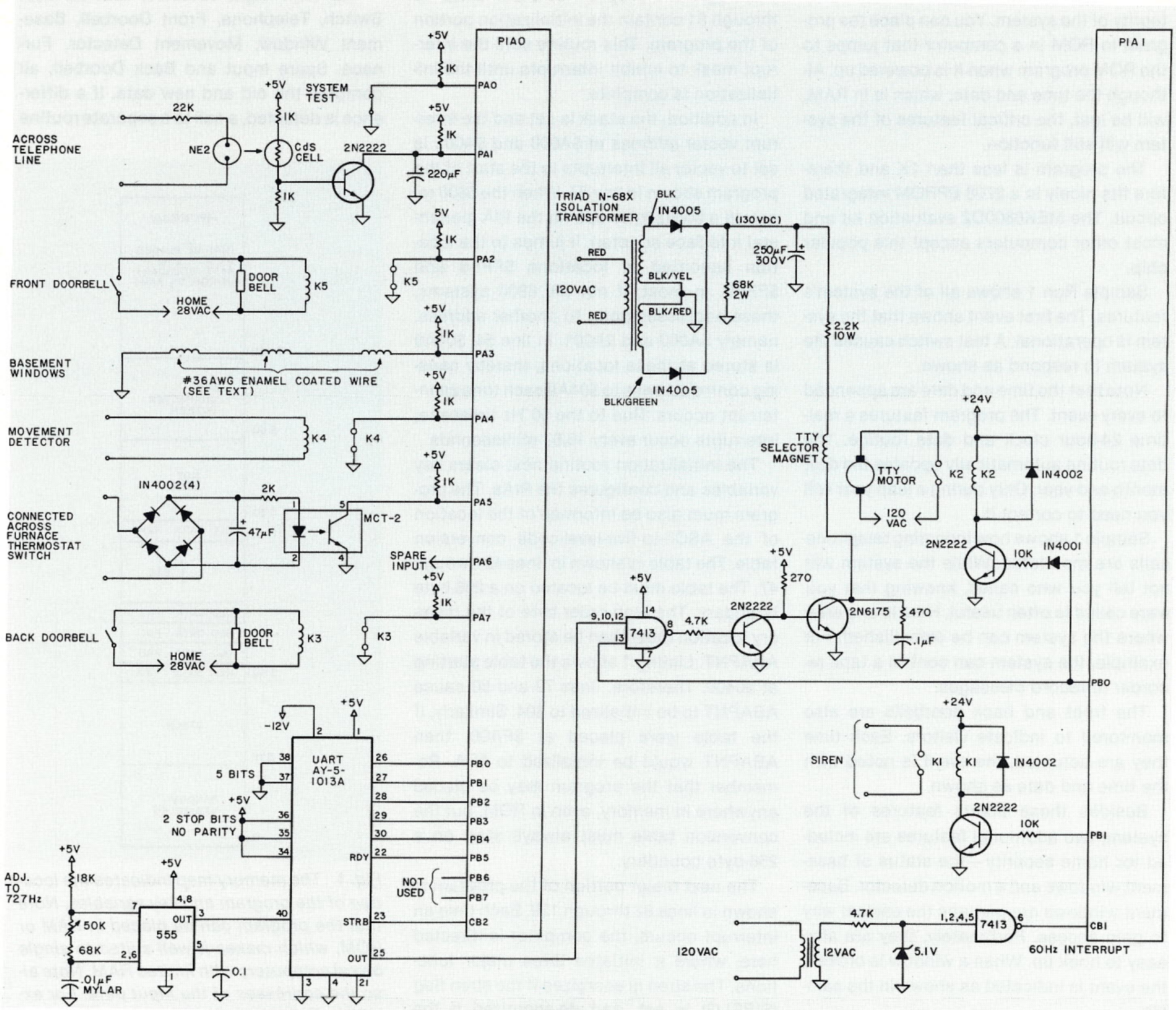
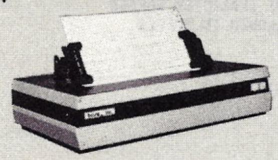


Fig. 2. Hardware necessary for compatibility with the software shown in Listing 1. The A port of PIA0 is used exclusively for the status inputs. The B port of the PIA is used to send text to the printer via a UART. Miscellaneous I/O, such as interrupts, siren and the printer motor, are controlled by the B port of PIA1.

PRINTERS & CRT'S From Orange Micro ✓310

APPLE GRAPHICS Interface Board Now Available

base 2, inc.



IMPACT PRINTER

$649.00

(LIST $699.00)

"The BASE 2 outperforms every printer in its price range. Do a comparison and see for yourself . . ."

★ GRAPHICS ★ TRACTORS / FRICTION FEED

• 2K Input Buffer • RS-232 Serial, Centronics® Parallel, IEEE-488, 20 ma • TRS-80 Cable option • 60 LPM - 100 CPS • Fast form feed • User programmable character set • 64, 72, 80, 96, 120, 132 Columns / line • Expanded characters • 9.5" wide paper • Automatic skip-over-perforation • Horizontal & Vertical tabs • Programmable vertical line spacing • Intel 8085 Microprocessor — over 40 software commands • Self test • 15 Baud rates to 9600 Baud • Optional foreign character sets

Interfaces to TRS-80, Apple, Atari, PET, Northstar, and most other computers.

TELEVIDEO CRT'S
PRICES SLASHED!

TVI 912C
TVI 920C
} Please Call Toll Free Prices are too low to advertise

PRINTERS

ANACOM 150 150 CPS, wide carriage, 9 x 9 dot . . . . . (List $1350) $ Call
CENTRONICS 737 Text processing dot matrix (Radio Shack LP IV) . . . . . $ Call
CENTRONICS 730 (Radio Shack Line Printer II) 639
COMPRINT 912 225 CPS Electrostatic . . . . . (List $660) 529
OKIDATA MICROLINE 80 . . . . . (List) $800) 599
EPSON Dot graphics, serial, parallel . . . . . $ Call
MALIBU Dot graphics, 132 Col, Letter quality . . . . . $ Call
PAPER TIGER IDS 440 w/graphics & 2K buffer . . . . . (List $1094) 939
QUME 5/45 Typewriter quality . . . . . (List $2905) 2499

INTERFACE EQUIPMENT

APPLE II — BASE 2 parallel graphics interface board 160
SSM AIO BOARD Serial/Parallel interface board . . . . . (List $225) 199
TRS-80 CABLES expansion interface or direct . . . . . $ Call

TOLL FREE **(800) 854-8275**
CA, AL, HI (714) 630-3322
Call for FREE CATALOG

Phone orders WELCOME. Same day shipment for VISA, MASTER CHARGE, and AMERICAN EXPRESS. Personal checks require 2 weeks to clear. Add 3% for shipping and handling. California residents add 6%. Manufacturer's warranty included. Prices subject to revision.

Orange Micro
3148 E. La Palma, Suite E
Anaheim, CA 92806

After you play the Temple of Apshai, you can play Sticks and Stones for free.

Within the 200 rooms and catacombs of the Temple of Apshai, untold treasures await you — the hero. All you have to do is elude, outsmart and outwit the beasts, monsters and demons lurking in the dark labyrinth. Spend minutes or hours on this role-playing fantasy — the boldest computer game in our Dunjonquest™ series.

Now, when you order the "Temple of Apshai," you get the "Sticks & Stones" board game for no extra charge. In fact, if you're not satisfied with the "Temple of Apshai," you can return it within 10 days and still keep "Sticks & Stones!"

But don't wait, this special offer is limited. (We'll also send you a catalog outlining our other exciting computer games).

Automated Simulations, Department KM ✓55
P.O. Box 4247, Mountain View, CA 94040

Please send me the "Temple of Apshai" for:

	Cassette ($24.95)	Disk ($29.95)
TRS-80	☐ 16K, Level II	☐ 32K TRSDOS
APPLE	Not available	☐ 48K Applesoft in ROM
PET	☐ 32K	Not available

(Add $1.00 shipping and handling charge; plus 6% or 6½% tax for California residents.)

Name ____________

Address ____________

City, State, Zip ____________

☐ Check enclosed. Charge to: ☐ VISA ☐ MasterCard

Amount $ ______ # ____________ Expiration date ______

Or charge by phone: (800) 824-7888, operator 861. In California: (800) 852-7777, operator 861. If you prefer, call these numbers for a list of the computer stores near you. N

mark the event.

The test switch is connected to PA0. Its primary purpose is to provide an indication that the system is functioning.

The telephone is connected to line PA1. When a phone call is received, the neon lamp (NE-2) lights, causing the photocell to change resistance and turn on the transistor. The capacitor is used simply to filter out the ring frequency. Note that the neon lamp/photocell combination must be packaged together and enclosed to ensure that external light does not falsely trigger the system.

The front doorbell and back doorbell are connected to PA2 and PA7, respectively. The interface circuits used are identical for each. The typical doorbell control circuit used in a home is merely a switch in series with a 28 V ac source connected across the doorbell. Each time the doorbell switch is depressed, 28 volts is applied to the doorbell coil. Fig. 2 shows a relay connected across the doorbell so that the relay will also be energized each time the doorbell switch is closed. The relay contact is then connected directly to the PIA.

The status of the furnace is detected through an optoisolator circuit. The circuit in Fig. 2 takes advantage of the fact that while the thermostat switch is open, there is 28 V ac across the switch. This voltage is merely rectified and applied to the optoisolator. The output, which is now TTL compatible, is connected to line PA5 of the PIA. Using this method causes current to flow through the furnace control relay while the thermostat switch is open. However, the current is far less than that needed to energize this relay.

Note that since the doorbell circuit is virtually identical to that of the furnace, you can interface the doorbells using an optoisolator circuit rather than using a relay as shown in Fig. 2.

The basement windows are connected to PA3. The status of windows can be detected in a variety of ways. For example, a small magnetic proximity switch and magnet may be attached to each window so that opening a window will open the circuit. Another method of detecting the status of a window is through a small glass breakage detector that is actuated by the vibration of glass breaking. Conductive foil or paint is yet another means of detection. The method shown consists of merely looping thin (#36AWG) enamel-coated wire through each window. When the window is removed, which is a virtual necessity to gain access through conventional basement windows, the circuit is broken.

The motion detector is connected to line PA4 of the PIA. These units are available commercially or can be built without too much difficulty. Most work on an ultrasonic principle. However, detecting changes in

Listing 1. Source listing for the computerized home security and status program. The program is written for the Motorola 6800 microprocessor.

```
8100           PIAODA  EQU  $8100     PIA OUTPUT,DATA DIRECTION REGISTERS
8101           PIAODB  EQU  $8101
8103           PIA1DB  EQU  $8103
8108           PIAOCA  EQU  $8108     PIA CONTROL,STATUS REGISTERS
8109           PIAOCB  EQU  $8109
810B           PIA1CB  EQU  $810B
003F           STACK   EQU  $3F       STACK
0040           DATA    EQU  $40       SAVE 32 BYTES FOR OLD AND NEW DATA
0060           PILE    EQU  $60       STARTING ADDRESS OF CHARACTER BUFFER
000A           MAXEVT  EQU  10        MAX # OF EVENTS ALLOWED IN BUFFER
0000                   ORG  0
0000           HR      RMB  1         HOUR COUNTER
0001           MIN     RMB  1         MINUTES COUNTER
0002           SEC     RMB  1         SECONDS COUNTER
0003           MON     RMB  1         MONTH COUNTER
0004           DAY     RMB  1         DAY COUNTER
0005           YR      RMB  1         YEAR COUNTER
0006           MAXDAY  RMB  1         MAXIMUM DAYS ALLOWED IN A GIVEN MONTH
               *
0007           FULAD1  RMB  2         TEMP. STORAGE LOCATION FOR INDEX REG.
0009           PILEAD  RMB  2         THIS ADDRESS RESEVERED FOR PILE POINTER
000B           PRTPIL  RMB  2         LOCATION OF ADDRESS OF CHARACTER TO BE PRINTED
000D           TEMPIR  RMB  2         SAVE X REG (ASCII BAUDOT ROUTINE)
000F           SAVTXT  RMB  2         SAVE X REG (CHR. BUFFER STORING ROUTINE)
0011           ABAPNT  RMB  2         POINTER FOR ASCII TO BAUDOT CONVERSION
0013           KBIT    RMB  1         USED TO CONTROL BIT LOOPING
0014           KBLOCK  RMB  1         LOCATION RESERVED TO COUNT # OF BLOCKS TO PRINT
0015           ONCE    RMB  1         CAUSES DATA TO SKIP OLD DATA AFTER FIRST TIME
0016           TOMUCH  RMB  1         FLAG TO INDICATE TO MUCH DATA IN FIFO
0017           KTIME   RMB  1         COUNTS INTERRUPTS
0018           CASET   RMB  1         CASE VARIABLE FOR ASCII TO BAUDOT CONVERSION
0019           FURFLG  RMB  1         FLAG INDICATES NEED TO APPEND FURNACE ELASPED TIME
001A           FURSEC  RMB  1         FURNACE SECONDS COUNTER
001B           FURMIN  RMB  1         FURNACE MINUTES COUNTER
001C           TTYTIM  RMB  1         TIME COUNTER FOR TELEPRINTER MOTOR TURN ON
001D           SIRTIM  RMB  1         COUNTER FOR NUMBER OF SECONDS SIREN IS ON
001E           SIRFLG  RMB  1         FLAG =0 SIREN OFF, IF 1 THERE IS A SIREN REQUEST
0400                   ORG  $0400
               ****************************************************************
               * ASCII TO BAUDOT CONVERSION TABLE, PLACE ON 256 BYTE BOUNDARY *
               ****************************************************************
0400 00                FCB  0,0,0,0,0,0,0,$25,0,0,2,0,0,8,0,0
0401 00 00
0403 00 00
0405 00 00
0407 25 00
0409 00 02
040B 00 00
040D 08 00
040F 00
0410 1F                FCB  $1F,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
0411 00 00
0413 00 00
0415 00 00
0417 00 00
0419 00 00
041B 00 00
041D 00 00
041F 00
0420 04                FCB  4,$2D,$31,$34,$29,0,$3A,$2B,$2F,$32,$2C,$23,$2C,$23,$3C,$3D
0421 2D 31
0423 34 29
0425 00 3A
0427 2B 2F
0429 32 2C
042B 23 2C
042D 23 3C
042F 3D
0430 36                FCB  $36,$37,$33,$21,$2A,$30,$35,$27,$26,$38,$2E,$3E,0,0,0,$39
0431 37 33
0433 21 2A
0435 30 35
0437 27 26
0439 38 2E
043B 3E 00
043D 00 00
043F 39
0440 00                FCB  0,3,$19,$0E,9,1,$0D,$1A,$14,6,$0B,$0F,$12,$1C,$0C,$18
0441 03 19
0443 0E 09
0445 01 0D
0447 1A 14
0449 06 0B
044B 0F 12
044D 1C 0C
044F 18
0450 16                FCB  $16,$17,$0A,5,$10,7,$1E,$13,$1D,$15,$11,0,$3D,0,0,0
0451 17 0A
0453 05 10
0455 07 1E
0457 13 1D
0459 15 11
045B 00 3D
045D 00 00
045F 00
               ****************************************************************
```

```
                    * INITIALIZE STACK, INTERRUPT VECTOR, AND CLEAR KEY VARIABLES *
                    * CONFIGURE PIA'S AND INITIALIZE REAL TIME CLOCK              *
                    ***************************************************************
0460 0F             BEGIN   SEI
0461 8E 00 3F               LDS    #STACK
0464 CE 04 A9               LDX    #START
0467 FF A0 00               STX    $A000
046A CE 00 60               LDX    #PILE      INITIALIZE PILE
046D DF 09                  STX    PILEAD
046F DF 0B                  STX    PRTPIL
0471 5F                     CLR B
0472 D7 14                  STA B  KBLOCK
0474 D7 15                  STA B  ONCE
0476 D7 16                  STA B  TOMUCH
0478 D7 17                  STA B  KTIME
047A D7 19                  STA B  FURFLG
047C D7 1E                  STA B  SIRFLG     START WITH SIREN REQUEST OFF
047E F7 81 08               STA B  PIA0CA     GET DATA DIRECTION REGISTERS
0481 F7 81 09               STA B  PIA0CB
0484 F7 81 0B               STA B  PIA1CB
0487 F7 81 00               STA B  PIA0DA     SET A PORT FOR ALL INPUTS
048A 86 1F                  LDA A  #$1F
048C B7 81 01               STA A  PIA0DB     SET B PORT FOR ALL OUTPUTS EXCEPT BIT 5,6,7
048F B7 81 03               STA A  PIA1DB     SET B PORT FOR ALL OUTPUTS EXCEPT BIT 5,6,7
0492 86 04                  LDA A  #4
0494 B7 81 08               STA A  PIA0CA     GET OUTPUT REGISTER
0497 B7 81 0B               STA A  PIA1CB     GET OUTPUT REGISTER
049A 86 2D                  LDA A  #$2D
049C B7 81 09               STA A  PIA0CB     CB1 IRQ ENABLE, CB2 OUTPUT ENABLE
049F F7 81 01               STA B  PIA0DB     ALL B PORT OUTPUTS LOW TTY MOTOR OFF
04A2 86 04                  LDA A  #$04
04A4 97 11                  STA A  ABAPNT     POINTER FOR ASCII TO BAUDOT CONVERSION
04A6 BD 06 EE               JSR    MTEST      INITIALIZE NUMBER OF DAYS IN MONTH
                    ***************************************************************
                    * EVERY 60 HZ INTERRUPT IS VECTORED HERE (START).  SIREN IS   *
                    * TURNED ON OR OFF AS NECESSARY.  REAL TIME CLOCK AND INPUTS  *
                    * TO SYSTEM ARE UPDATED.                                      *
                    ***************************************************************
04A9 B6 81 01       START   LDA A  PIA0DB     CLEAR INTERRUPT
04AC 96 1E                  LDA A  SIRFLG     IS THERE A NEED TO TURN SIREN ON
04AE 27 2B                  BEQ    NOSIR
04B0 B6 81 03               LDA A  PIA1DB     GET PRESENT STATUS
04B3 85 02                  BIT A  #2         IS SIREN ALREADY ON
04B5 26 10                  BNE    SIREN
04B7 8A 02                  ORA A  #2
04B9 B7 81 03               STA A  PIA1DB     TURN ON SIREN
04BC CE 07 E1               LDX    #SIRON
04BF BD 05 F5               JSR    STORE
04C2 7F 00 1D               CLR    SIRTIM     CLEAR SIREN SECOND COUNTER
04C5 20 14                  BRA    NOSIR
04C7 D6 1D          SIREN   LDA B  SIRTIM
04C9 C1 96                  CMP B  #150       IS SIREN ON FOR 2.5 MINUTES
04CB 26 0E                  BNE    NOSIR
04CD 84 FD                  AND A  #$FD
04CF B7 81 03               STA A  PIA1DB     TURN OFF SIREN
04D2 CE 07 EA               LDX    #SIROF
04D5 BD 05 F5               JSR    STORE
04D8 7F 00 1E               CLR    SIRFLG
04DB BD 06 8B       NOSIR   JSR    TIMESR
04DE 96 16                  LDA A  TOMUCH
04E0 27 01                  BEQ    COTIN
04E2 3B                     RTI               RETURN IF THERE IS TO MUCH DATA IN FIFO
04E3 CE 00 40       COTIN   LDX    #DATA      STORING POINTER
04E6 B6 81 00               LDA A  PIA0DA
                    * STORE INPUT STATUS IN BYTE FORM *
04E9 7F 00 13               CLR    KBIT       CLEAR BIT COUNTER
04EC 47             AGAIN   ASR A
04ED 24 04                  BCC    PLUS
04EF C6 01                  LDA B  #1
04F1 20 01                  BRA    STABIT
04F3 5F             PLUS    CLR B
04F4 E7 00          STABIT  STA B  0,X
04F6 08                     INX
04F7 7D 00 15               TST    ONCE
04FA 26 02                  BNE    NOTFST
04FC E7 00                  STA B  0,X
04FE 08             NOTFST  INX
04FF 7C 00 13               INC    KBIT
0502 D6 13                  LDA B  KBIT
0504 C1 08                  CMP B  #8
0506 26 E4                  BNE    AGAIN
                    *********************
                    * TEST SWITCH INPUT *
                    *********************
0508 CE 00 40               LDX    #DATA
050B A6 00                  LDA A  0,X        GET NEW DATA
050D E6 01                  LDA B  1,X        GET OLD DATA
050F 11                     CBA
0510 27 0A                  BEQ    SAME0
0512 A7 01                  STA A  1,X        STORE NEW DATA IN OLD DATA LOCATION
0514 27 06                  BEQ    SAME0
0516 CE 07 54               LDX    #TESTON    GET TEST ON TEXT
0519 BD 05 F5               JSR    STORE
                    *********************
                    * TELEPHONE INPUT   *
                    *********************
051C A6 02          SAME0   LDA A  2,X        GET NEW DATA
051E E6 03                  LDA B  3,X        GET OLD DATA
0520 11                     CBA               IS THE OLD DATA AND NEW DATA THE SAME
0521 27 0A                  BEQ    SAME1
0523 A7 03                  STA A  3,X        STORE NEW DATA IN OLD DATA LOCATION
```

the ambient light level is another method that can be used. See the references for an article by Walter Gontowski describing this method in detail.

The advantage of a motion detector lies in its ability to cover an entire area or room. Most of these units have an accessory 120 V ac controlled output to allow the unit to turn on a light or some other device. Fig. 2 merely shows a 120 V ac relay connected to such as output. The relay contact is then connected to the PIA. Here again, an opto-isolator circuit could be used.

The B port of PIA0 is used exclusively to send characters to the printer. An ACIA (asynchronous communications interface adapter) integrated circuit is typically used to interface a computer to a serial output. However, the ACIA does not handle a five-bit code such as Baudot.

In addition, some evaluation kits such as the MEK6800D2 do not have an unused ACIA. Therefore, a UART provides the necessary serial output. Of the seven lines interconnecting the UART and PIA, five are devoted to encoding the character, one is a handshaking signal to indicate the ready status of the UART and one is to strobe (initiate) the transmission of the character.

Only three lines of PIA1 are used. The printer motor and the 60 milliampere current loop for the selector magnet are both controlled by the PB0 output of PIA1. The motor is controlled to prevent unnecessary wear and tear on the printer. The current loop is opened to prevent wasting energy.

The siren is controlled by the PB1 output. The relay contact shown in Fig. 2 is one simple way to interface the PIA to the siren. A relay contact is shown since it is a "universal" output capable of interfacing to almost anything. Other interface methods may be used at the discretion of the user.

The 60 Hz interrupt signal is connected to CB1. The 12 V ac output of the transformer is clipped at 5.1 volts by the zener diode and shaped into a square wave by the 7413 Schmitt trigger.

Conclusion

The possibilities for a computerized home security and status system are virtually endless. Smoke detectors can be added to save a home from a possible fire. A telephone calling subsystem can be added to automatically call the fire or police departments. A fluid detector can be added to trigger an alarm in the event that a basement is about to be flooded.

This article would have been unreasonably long if it had included circuits for each of these subsystems. In addition, periodicals abound with such circuits. For those needing schematics or ideas for some of these circuits mentioned, I have included some references for further reading.

I hope that the security portion of the system will never have to be tested in a real-life situation at your home. However, it it is needed, it is nice to know that it is there. ■

References

Ciarcia, Steven. "Build a Computer Controlled Security System for Your Home: Part 1." *Byte* (January 1979), 56.

Ciarcia, Steven. "Build a Computer Controlled Security System for Your Home: Part 2," *Byte* (February 1979), 162.

Ciarcia, Steven. "Build a Computer Controlled Security System for Your Home: Part 3," *Byte* (March 1979), 150.

Ciarcia, Steven. "Computerize a Home," *Byte* (January 1980), 28.

Gontowski, Walter. "Motion-Detection Alarm," *Popular Electronics* (May 1980), 61.

Hollabaugh, John. "A Portable Alarm for Single Entries," *Popular Electronics* (December 1978), 69.

Olson, Hank. "An Infrared Intrusion System," *Popular Electronics* (December 1978), 66.

Shambaugh, David E. "Home Applications for the 6800," *Interface Age* (June 1979), 68.

Trollope, Gregory, A. R. "Do You Need the Real Time," *Byte* (November 1977), 166.

Turner, Bill. "The House Gimix Built," *Interface Age* (June 1979), 57.

Wierenga, Theron. "A Furnace Watchdog," *Byte* (January 1980), 74.

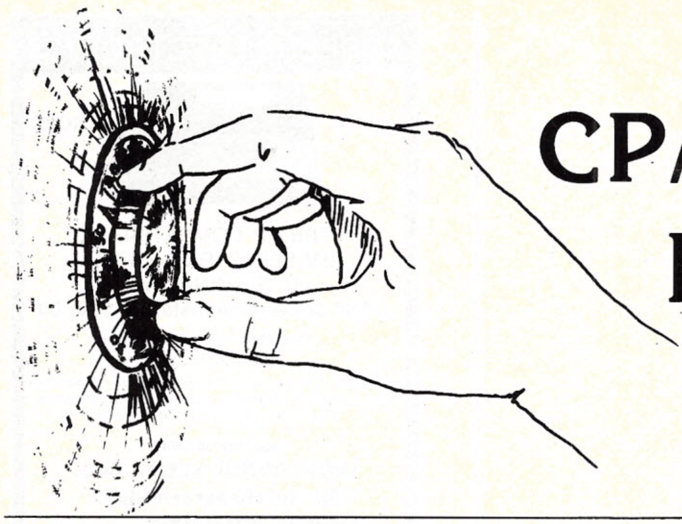

```
0525 27 06              BEQ    SAME1
0527 CE 07 59           LDX    #PHONE
052A BD 05 F5           JSR    STORE
                 *********************
                 * FRONT DOORBELL    *
                 *********************
052D A6 04       SAME1  LDA A  4,X      GET NEW DATA
052F E6 05              LDA B  5,X      GET OLD DATA
0531 11                 CBA
0532 27 0A              BEQ    SAME2
0534 A7 05              STA A  5,X      STORE NEW DATA IN OLD DATA LOCATION
0536 27 06              BEQ    SAME2    IF NEW DATA IS ZERO DO NOT STORE TEXT
0538 CE 07 63           LDX    #FDOOR   GET FRONT DOOR TEXT
053B BD 05 F5           JSR    STORE
                 *********************
                 * BASEMENT WINDOW   *
                 *********************
053E A6 06       SAME2  LDA A  6,X      GET NEW DATA
0540 E6 07              LDA B  7,X      GET OLD DATA
0542 11                 CBA
0543 27 16              BEQ    SAME3    BRANCH IF NO CHANGE
0545 A7 07              STA A  7,X      STORE NEW DATA IN OLD DATA LOCATION
0547 27 0C              BEQ    BASOFF
0549 CE 07 72           LDX    #BASEON  BASEMENT WINDOW BROKEN TEXT
054C BD 05 F5           JSR    STORE
054F 86 01              LDA A  #1
0551 97 1E              STA A  SIRFLG   SET SIREN FLAG FOR REQUEST
0553 20 06              BRA    SAME3
0555 CE 07 89    BASOFF LDX    #BASEOF  BASEMENT WINDOW COMPLETE TEXT
0558 BD 05 F5           JSR    STORE
                 *********************
                 * MOVEMENT DETECTOR *
                 *********************
055B A6 08       SAME3  LDA A  8,X      GET NEW DATA
055D E6 09              LDA B  9,X      GET OLD DATA
055F 11                 CBA
0560 27 0E              BEQ    SAME4
0562 A7 09              STA A  9,X      STORE NEW DATA IN OLD DATA LOCATION
0564 27 0A              BEQ    SAME4
0566 CE 07 A2           LDX    #MOVEON  GET MOVEMENT ON TEXT
0569 BD 05 F5           JSR    STORE
056C 86 01              LDA A  #1
056E 97 1E              STA A  SIRFLG   SET SIREN FLAG FOR A REQUEST
                 *********************
                 * FURNACE           *
                 *********************
0570 A6 0A       SAME4  LDA A  10,X     GET NEW DATA
0572 E6 0B              LDA B  11,X     GET OLD DATA
0574 11                 CBA
0575 27 1B              BEQ    SAME5    BRANCH IF NO CHANGE
0577 A7 0B              STA A  11,X     STORE NEW DATA IN OLD DATA LOCATION
0579 27 0E              BEQ    FUROFF
057B CE 07 B2           LDX    #FURNON  GET FURNACE ON TEXT
057E BD 05 F5           JSR    STORE
0581 7F 00 1A           CLR    FURSEC   INITIALIZE SECONDS COUNTER
0584 7F 00 1B           CLR    FURMIN   INITIALIZE MINUTES COUNTER
0587 20 09              BRA    SAME5
0589 7C 00 19    FUROFF INC    FURFLG   SET FURNACE ELASPED TIME FLAG FOR PRINTING
058C CE 07 BD           LDX    #FURNOF  GET FURNACE OFF TEXT
058F BD 05 F5           JSR    STORE
                 *********************
                 * SPARE INPUT       *
                 *********************
0592 01          SAME5  NOP
                 *********************
                 * BACK DOORBELL     *
                 *********************
0593 A6 0E       SAME6  LDA A  14,X     GET NEW DATA
0595 E6 0F              LDA B  15,X     GET OLD DATA
0597 11                 CBA
0598 27 0A              BEQ    SCANOV   BRANCH IF NO CHANGE
059A A7 0F              STA A  15,X     STORE NEW DATA IN OLD DATA LOCATION
059C 27 06              BEQ    SCANOV   IF NEW DATA IS ZERO, DO NOT STORE TEXT
059E CE 07 D3           LDX    #RDOOR   GET REAR DOOR TEXT
05A1 BD 05 F5           JSR    STORE
05A4 96 15       SCANOV LDA A  ONCE     IF IT IS ZERO, IT IS FIRST TIME THROUGH
05A6 27 01              BEQ    FIRSTT   THEREFORE IT IS NOT FROM AN INTERRUPT REQUEST
05A8 3B                 RTI
05A9 7C 00 15    FIRSTT INC    ONCE     FLAG TO INDICATE FIRST RUN THROUGH
05AC 0E                 CLI
                 ***************************************************************
                 * WHILE WAITING FOR 60 HZ INTERRUPT CHECK FOR EVENTS TO PRINT *
                 * AND PRINT IF NECESSARY.  TTY MOTOR AND 60 MA LOOP ARE       *
                 * ENERGIZED BEFORE PRINTING.  ASCII CHARACTERS ARE CONVERTED  *
                 * TO BAUDOT BEFORE PRINTING.                                  *
                 ***************************************************************
05AD 96 14       NODATA LDA A  KBLOCK   IS THERE ANYTHING TO PRINT
05AF 27 FC              BEQ    NODATA   CONTINUE LOOPING IF THERE IS NOTHING TO PRINT
                 *                          THERE IS SOMETHING TO PRINT
05B1 B6 81 03           LDA A  PIA1DB   GET PRESENT STATE OF B PORT
05B4 88 01              EOR A  #$01     CHANGE STATE OF BIT 0
05B6 B7 81 03           STA A  PIA1DB   TURN ON TTY MOTOR AND 60 MA LOOP
05B9 7F 00 1C           CLR    TTYTIM   INITIALIZE TELEPRINTER MOTOR TURN ON COUNTER
05BC 96 1C       NOTRDY LDA A  TTYTIM
05BE 81 03              CMP A  #3       WAIT 3 SECONDS BEFORE BEGINNING TO PRINT
05C0 26 FA              BNE    NOTRDY
05C2 B6 81 01    PRINT  LDA A  PIA0DB   IS UART READY FOR ANOTHER CHARACTER
05C5 84 20              AND A  #$20     CHECK BIT 5
05C7 27 F9              BEQ    PRINT
05C9 DE 0B              LDX    PRTPIL   SINCE ACIA IS READY GET ADDR. OF LTR. TO PR
```

```
05CB A6 00            LDA A  X          GET THE NEXT CHARACTER TO BE PRINTED
05CD 81 04            CMP A  #$04       END OF LINE DELIMITER
05CF 27 08            BEQ    DONE1      BRANCH IF IT IS THE END OF THE BLOCK
05D1 BD 07 13         JSR    ASCBAU     CONVERT CHARACTER TO BAUDOT
05D4 08        NOPRT  INX
05D5 DF 0B            STX    PRTPIL     SAVE ADDRESS OF NEXT CHARACTER TO BE PRINTED
05D7 20 E9            BRA    PRINT
05D9 08        DONE1  INX               UPDATE THE ADDRESS OF NEXT CHARACTER TO BE PRINTED
05DA DF 0B            STX    PRTPIL     STORE ADDRESS OF NEXT CHARACTER TO BE PRINTED
05DC 7A 00 14         DEC    KBLOCK     DECREMENT THE BLOCK COUNTER
05DF 26 E1            BNE    PRINT
               * DONE PRINTING, INITIALIZE PILE AND PRINT ADDRESS, TURN OFF TTY
05E1 B6 81 03         LDA A  PIA1DB     GET PRESENT SATE OF B PORT
05E4 88 01            EOR A  #$01       CHANGE STATE OF BIT 0
05E6 B7 81 03         STA A  PIA1DB     TURN OFF TTY MOTOR AND 60 MA LOOP
05E9 CE 00 60         LDX    #PILE
05EC DF 09            STX    PILEAD
05EE DF 0B            STX    PRTPIL
05F0 7F 00 16         CLR    TOMUCH     CLEAR TO MUCH DATA IN FIFO FLAG
05F3 20 B8            BRA    NODATA

               **********************************************
               * STORE EVENT AND APPEND TIME, DATE, AND EOL *
               **********************************************
05F5 96 16     STORE  LDA A  TOMUCH     DO NOT STORE EVENT IF CHR. BUFFER IS FULL
05F7 26 6F            BNE    NEXTWD
05F9 DF 0F     NEXT4  STX    SAVTXT     SAVE X-REG POINTER OF EVENT TEXT
05FB D6 00            LDA B  HR
05FD 8D 72            BSR    STOTIM
05FF 86 3A            LDA A  #':
0601 8D 66            BSR    STOCHR
0603 D6 01            LDA B  MIN
0605 8D 6A            BSR    STOTIM
0607 86 3A            LDA A  #':
0609 8D 5E            BSR    STOCHR
060B D6 02            LDA B  SEC
060D 8D 62            BSR    STOTIM
060F 86 20            LDA A  #$20
0611 8D 56            BSR    STOCHR
0613 8D 54            BSR    STOCHR
0615 D6 03            LDA B  MON
0617 8D 58            BSR    STOTIM
0619 86 2F            LDA A  #'/
061B 8D 4C            BSR    STOCHR
061D D6 04            LDA B  DAY
061F 8D 50            BSR    STOTIM
0621 86 2F            LDA A  #'/
0623 8D 44            BSR    STOCHR
0625 D6 05            LDA B  YR
0627 8D 48            BSR    STOTIM
0629 86 20            LDA A  #$20
062B 8D 3C            BSR    STOCHR
062D DE 0F            LDX    SAVTXT     RETRIEVE X-REG POINTER OF EVENT TEXT
062F 8D 28            BSR    STOR       STORE TEXT
0631 96 19            LDA A  FURFLG     APPEND FURNACE ELASPED TIME ?
0633 27 0F            BEQ    NOFURN
0635 D6 1B            LDA B  FURMIN     GET MINUTES FURNACE IS ON
0637 8D 38            BSR    STOTIM
0639 86 3A            LDA A  #':
063B 8D 2C            BSR    STOCHR
063D D6 1A            LDA B  FURSEC     GET SECONDS FURNACE IS ON
063F 8D 30            BSR    STOTIM
0641 7F 00 19         CLR    FURFLG     RESET FURNACE PRINT ELASPED TIME FLAG
0644 CE 07 F4  NOFURN LDX    #EOL       GET END OF LINE CHARACTERS
0647 8D 10            BSR    STOR       STORE EOL
0649 CE 00 40         LDX    #DATA
064C 7C 00 14         INC    KBLOCK
064F 96 14            LDA A  KBLOCK     CHECK TO SEE IF FIFO IS FULL
0651 81 0A            CMP A  #MAXEVT
0653 25 13            BCS    NEXTWD     BRANCH IF CHARACTER BUFFER FULL
0655 7C 00 16         INC    TOMUCH     SET TOMUCH FLAG , FIFO IS FULL
0658 39               RTS
               *
0659 A6 00     STOR   LDA A  0,X        GET A CHARACTER FROM THE DICTIONARY
065B 81 2A            CMP A  #'*        ARE WE AT THE END OF THE WORD
065D 27 09            BEQ    NEXTWD     BRANCH IF AT END OF WORD
065F 08               INX               GET READY FOR NEXT CHARACTER
0660 DF 07            STX    FULAD1
0662 8D 05            BSR    STOCHR     STORE A SINGLE CHARACTER FROM ACC A
0664 DE 07            LDX    FULAD1
0666 20 F1            BRA    STOR
0668 39        NEXTWD RTS
               *
0669 DE 09     STOCHR LDX    PILEAD
066B A7 00            STA A  0,X        STORE A CHARACTER
066D 08               INX
066E DF 09            STX    PILEAD
0670 39               RTS
               ***********************************************************
               * CONVERT A BINARY NUMBER IN ACC B TO TWO ASCII CHARACTERS *
               * AND STORE IN CHARACTER BUFFER.                           *
               ***********************************************************
0671 4F        STOTIM CLR A             A HOLDS HIGH ORDER DIGIT
0672 C1 0A     OUT1   CMP B  #10        B>9?
0674 2B 06            BMI    OUT2       DONE IF NOT
0676 8B 10            ADD A  #$10       A=A+$10
0678 C0 0A            SUB B  #10        B=B-10
067A 20 F6            BRA    OUT1       LOOP
067C 1B        OUT2   ABA
067D 44               LSR A
067E 44               LSR A
```

```
067F 44                     LSR A
0680 44                     LSR A
0681 8A 30                  ORA A  #$30
0683 8D E4                  BSR    STOCHR    STORE HIGH ORDER NUMBER
0685 CA 30                  ORA B  #$30
0687 17                     TBA
0688 8D DF                  BSR    STOCHR    STORE LOW ORDER NUMBER
068A 39                     RTS
                    **************************
                    * REAL TIME CLOCK ROUTINE *
                    **************************
068B 7C 00 17       TIMESR  INC    KTIME     ADD ONE TO INTERRUPT COUNTER
068E 96 17                  LDA A  KTIME
0690 81 3C                  CMP A  #60       WAIT FOR 60 INTERRUPTS
0692 27 01                  BEQ    CLOCK     IF 60 INTERRUPTS PASSED GO TO CLOCK PROGRAM
0694 39                     RTS
                    *
0695 5F             CLOCK   CLR B
0696 D7 17                  STA B  KTIME
0698 7C 00 1C               INC    TTYTIM    INCREMENT TELEPRINTER MOTOR TURN ON COUNTER
069B 7C 00 1D               INC    SIRTIM    ADD ONE TO SIREN TIME COUNTER
069E 7C 00 1A               INC    FURSEC    INCREMENT FURNACE COUNTERS
06A1 96 1A                  LDA A  FURSEC
06A3 81 3C                  CMP A  #60
06A5 26 05                  BNE    INCSEC
06A7 D7 1A                  STA B  FURSEC
06A9 7C 00 1B               INC    FURMIN
06AC 7C 00 02       INCSEC  INC    SEC       UPDATE SECONDS
06AF 96 02                  LDA A  SEC
06B1 81 3C                  CMP A  #60
06B3 26 5D                  BNE    DONE
06B5 D7 02                  STA B  SEC
06B7 7C 00 01               INC    MIN       UPDATE MINUTES
06BA 96 01                  LDA A  MIN
06BC 81 3C                  CMP A  #60
06BE 26 52                  BNE    DONE
06C0 D7 01                  STA B  MIN
06C2 7C 00 00               INC    HR        UPDATE HOURS
06C5 96 00                  LDA A  HR
06C7 81 18                  CMP A  #24
06C9 26 47                  BNE    DONE
06CB D7 00                  STA B  HR
06CD 7C 00 04               INC    DAY       UPDATE DAYS
06D0 96 04                  LDA A  DAY
06D2 91 06                  CMP A  MAXDAY
06D4 26 3C                  BNE    DONE
06D6 86 01                  LDA A  #1        SET TO FIRST DAY OF MONTH
06D8 97 04                  STA A  DAY
06DA 8D 12                  BSR    MTEST
06DC 7C 00 03               INC    MON       UPDATE MONTH
06DF 96 03                  LDA A  MON
06E1 81 0D                  CMP A  #13
06E3 26 2D                  BNE    DONE
06E5 86 01                  LDA A  #1        SET TO FIRST MONTH OF NEW YEAR
06E7 97 03                  STA A  MON
06E9 7C 00 05               INC    YR
06EC 20 24                  BRA    DONE
06EE 96 04          MTEST   LDA A  DAY
06F0 81 02                  CMP A  #2
06F2 27 15                  BEQ    MO28      SET TO MONTH WITH 28 DAYS
06F4 81 04          APRIL   CMP A  #4
06F6 27 16                  BEQ    MO30      SET TO MONTH WITH 30 DAYS
06F8 81 06          JUNE    CMP A  #6
06FA 27 12                  BEQ    MO30      SET TO MONTH WITH 30 DAYS
06FC 81 09          SEPT    CMP A  #9
06FE 27 0E                  BEQ    MO30      SET TO MONTH WITH 30 DAYS
0700 81 0B          NOV     CMP A  #11
0702 27 0A                  BEQ    MO30
0704 86 20          MO31    LDA A  #32       IF NO MATCH ASSUME A 31 DAY MONTH
0706 97 06                  STA A  MAXDAY
0708 39                     RTS
0709 86 1D          MO28    LDA A  #29       INITIALIZE FOR A 28 DAY MONTH
070B 97 06                  STA A  MAXDAY
070D 39                     RTS
070E 86 1F          MO30    LDA A  #31       INITIALIZE FOR A 30 DAY MONTH
0710 97 06                  STA A  MAXDAY
0712 39             DONE    RTS
                    ***********************************************
                    * ASCII TO BAUDOT PRINT AND CONVERSION ROUTINE *
                    ***********************************************
0713 DF 0D          ASCBAU  STX    TEMPIR    SAVE INDEX REGISTER
0715 97 12                  STA A  ABAPNT+1
0717 DE 11                  LDX    ABAPNT
0719 A6 00                  LDA A  0,X       GET A BAUDOT CHARACTER
071B 36                     PSH A
071C 81 82                  CMP A  #$82      IF LF DO NOT CHANGE CASE
071E 27 2D                  BEQ    PRINTB
0720 81 84                  CMP A  #$84      IF SPACE DO NOT CHANGE CASE
0722 27 29                  BEQ    PRINTB
0724 81 88                  CMP A  #$88      IF CR DO NOT CHANGE CASE
0726 27 25                  BEQ    PRINTB
0728 16                     TAB              TRANSFER A TO B FOR TESTS
0729 C4 20                  AND B  #$20      ISOLATE BIT 5
072B D8 18                  EOR B  CASET     HAS THERE BEEN A CASE CHANGE
072D 27 1E                  BEQ    PRINTB    IF NO CHANGE GO PRINT A CHARACTER
072F 85 20                  BIT A  #$20      DO WE SEND LTRS OR FIGS
0731 26 0A                  BNE    FIGHR     GO SEND A FIGS CHARACTER
0733 7F 00 18               CLR    CASET     SET CASE FOR NEXT TIME
0736 86 1F                  LDA A  #$1F      LDA WITH LTRS CHARACTER
0738 B7 81 01               STA A  PIAODB    PRINT A CHARACTER
```

```
     073B 20 09             BRA    PRINT1
     073D 86 20     FIGHR   LDA A  #$20
     073F 97 18             STA A  CASET      SET CASE FOR NEXT TIME
     0741 86 1B             LDA A  #$1B       LDA
     0743 B7 81 01          STA A  PIAODB     PRINT A CHARACTER
     0746 B6 81 01  PRINT1  LDA A  PIAODB     IS UART READY FOR ANOTHER CHARACTER
     0749 84 20             AND A  #$20       IS BIT 5 HIGH
     074B 27 F9             BEQ    PRINT1
     074D 32        PRINTB  PUL A             RETURN BAUDOT CHARACTER TO ACC A
     074E B7 81 01          STA A  PIAODB     PRINT A CHARACTER
     0751 DE 0D     NOMTCH  LDX    TEMPIR     RETURN INDEX REG.
     0753 39                RTS
     0754 54        TESTON  FCC    /TEST*/
     0755 45 53
     0757 54 2A
     0759 54        PHONE   FCC    /TELEPHONE*/
     075A 45 4C
     075C 45 50
     075E 48 4F
     0760 4E 45
     0762 2A
     0763 46        FDOOR   FCC    /FRONT DOORBELL*/
     0764 52 4F
     0766 4E 54
     0768 20 44
     076A 4F 4F
     076C 52 42
     076E 45 4C
     0770 4C 2A
     0772 42        BASEON  FCC    /BASEMENT WINDOW BROKEN*/
     0773 41 53
     0775 45 4D
     0777 45 4E
     0779 54 20
     077B 57 49
     077D 4E 44
     077F 4F 57
     0781 20 42
     0783 52 4F
     0785 4B 45
     0787 4E 2A
     0789 42        BASEOF  FCC    /BASEMENT WINDOW COMPLETE*/
     078A 41 53
     078C 45 4D
     078E 45 4E
     0790 54 20
     0792 57 49
     0794 4E 44
     0796 4F 57
     0798 20 43
     079A 4F 4D
     079C 50 4C
     079E 45 54
     07A0 45 2A
     07A2 4D        MOVEON  FCC    /MOTION DETECTED*/
     07A3 4F 54
     07A5 49 4F
     07A7 4E 20
     07A9 44 45
     07AB 54 45
     07AD 43 54
     07AF 45 44
     07B1 2A
     07B2 46        FURNON  FCC    /FURNACE ON*/
     07B3 55 52
     07B5 4E 41
     07B7 43 45
     07B9 20 4F
     07BB 4E 2A
     07BD 46        FURNOF  FCC    /FURNACE ELASPED TIME */
     07BE 55 52
     07C0 4E 41
     07C2 43 45
     07C4 20 45
     07C6 4C 41
     07C8 53 50
     07CA 45 44
     07CC 20 54
     07CE 49 4D
     07D0 45 20
     07D2 2A
     07D3 42        RDOOR   FCC    /BACK DOORBELL*/
     07D4 41 43
     07D6 4B 20
     07D8 44 4F
     07DA 4F 52
     07DC 42 45
     07DE 4C 4C
     07E0 2A
     07E1 53        SIRON   FCC    /SIREN ON*/
     07E2 49 52
     07E4 45 4E
     07E6 20 4F
     07E8 4E 2A
     07EA 53        SIROF   FCC    /SIREN OFF*/
     07EB 49 52
     07ED 45 4E
     07EF 20 4F
     07F1 46 46
     07F3 2A
```

```
473 07F4 0A        EOL      FCB      $0A,$0A,$0D,$0D,$04,'*
    07F5 0A 0D
    07F7 0D 04
    07F9 2A
474                         END      BEGIN

NO ERROR(S) DETECTED

  SYMBOL TABLE:

ABAPNT 0011   AGAIN  04EC   APRIL  06F4   ASCBAU 0713   BASEOF 0789
BASEON 0772   BASOFF 0555   BEGIN  0460   CASET  0018   CLOCK  0695
COTIN  04E3   DATA   0040   DAY    0004   DONE   0712   DONE1  05D9
EOL    07F4   FDOOR  0763   FIGHR  073D   FIRSTT 05A9   FULAD1 0007
FURFLG 0019   FURMIN 001B   FURNOF 07BD   FURNON 07B2   FUROFF 0589
FURSEC 001A   HR     0000   INCSEC 06AC   JUNE   06F8   KBIT   0013
KBLOCK 0014   KTIME  0017   MAXDAY 0006   MAXEVT 000A   MIN    0001
MO28   0709   MO30   070E   MO31   0704   MON    0003   MOVEON 07A2
MTEST  06EE   NEXT4  05F9   NEXTWD 0668   NODATA 05AD   NOFURN 0644
NOMTCH 0751   NOPRT  05D4   NOSIR  04DB   NOTFST 04FE   NOTRDY 05BC
NOV    0700   ONCE   0015   OUT1   0672   OUT2   067C   PHONE  0759
PIAOCA 8108   PIAOCB 8109   PIAODA 8100   PIAODB 8101   PIA1CB 810B
PIA1DB 8103   PILE   0060   PILEAD 0009   PLUS   04F3   PRINT  05C2
PRINT1 0746   PRINTB 074D   PRTPIL 000B   RDOOR  07D3   SAMEO  051C
SAME1  052D   SAME2  053E   SAME3  055B   SAME4  0570   SAME5  0592
SAME6  0593   SAVTXT 000F   SCANOV 05A4   SEC    0002   SEPT   06FC
SIREN  04C7   SIRFLG 001E   SIROF  07EA   SIRON  07E1   SIRTIM 001D
STABIT 04F4   STACK  003F   START  04A9   STOCHR 0669   STOR   0659
STORE  05F5   STOTIM 0671   TEMPIR 000D   TESTON 0754   TIMESR 068B
TOMUCH 0016   TTYTIM 001C   YR     0005
```



CP/M Encryption Prescription

A software cipher will ensure privacy for your files—and peace of mind for you.

Alan Sclawy
1119 Avenue I
Brooklyn, NY 11230

You hear a lot about data encryption these days—ways of scrambling confidential files so they can't be read by intruders. There is now a data encryption standard (DES); a chip implementing this standard; and lots of proposals for alternative systems, including some very attractive "public-key" systems.

But many of us can't use these sophisticated techniques. In my case, for example, I work in an academic time-sharing environment; the computer center would take a dim view if I tried to install a DES chip as a peripheral. Still, schools are notorious for characters who regard the computer, its users and their files as fair game. It was important to be able to edit, format and store sensitive materials, like examinations and class records. So I devised this simple way to frustrate snoopers.

The program is easy to implement and convenient to use, yet it is based on sound and well-understood cryptographic techniques. Although files encrypted with this program would probably yield their secrets to a determined professional in a few hours, they will be secure enough to thwart the amateur, which is all most of us usually need.

The version printed here is written in BASIC for use under CP/M, but I have written other versions in Fortran and in C and found them equally satisfactory.

Before describing the program, I want to define a few terms. *Encryption* is any kind of transformation of a text to make it unreadable except by authorized individuals. A *cipher* is an encryption method that transforms the text letter by letter, and to *encipher* a text is to apply a cipher to it. (That's what my program does.) *Plaintext* is the message to be encrypted (or enciphered); *ciphertext* is the enciphered version of the message. To *decrypt* a message is to transform it back so it can be read, and to *decipher* is to decrypt an enciphered message. *Cryptanalysis* is analyzing and cracking an encryption technique by studying the material transmitted. It's usually assumed that an authorized recipient deciphers a message, while an unauthorized individual resorts to cryptanalysis.

The Method

The encryption method used here was devised by Gilbert S. Vernam, an engineer at AT&T, in 1917. That was a long time before the computer era, but his techniques will look very familiar to anyone who has been around computers. Vernam's algorithm has been described in great detail by David Kahn in his classic book, *The Codebreakers*. Briefly, it consists of the following steps:

1. Encode the message as a bit string.
2. Generate a random bit string of the same length as the message. (This is the key.)
3. Execute an exclusive-OR (XOR) between corresponding bits of the message and the key. The output bit string is the ciphertext.

To decipher the message, you apply the

Plaintext:	H	e	l	l	o
Plaintext (ASCII):	01001000	01100101	01101100	01101100	01101111
Key:	10010101	11000100	00110101	11011010	01010110
Ciphertext:	11011101	10100001	01011001	10110110	00111001

Fig. 1. Example of text enciphered by Vernam's method.

p	q	p XOR q
0	0	0
0	1	1
1	0	1
1	1	0

Fig. 2. Table of the exclusive-OR (XOR) function.

```
(A)  [B]O Y B O Y B O Y B O Y[B]O Y B
     [G]I R L G I R L G I R L[G]I R L

(B)  [P]A M E L A P A M E L A[P]A M E
     [J]O H N J O H N J O H N[J]O H N
```

*Fig. 3. Getting long key periods from short keywords. (a) Key lengths have no common factors; period = (3*4) = 12. (b) Key lengths have a common factor of 2; period shorter than maximum.*

same key to the ciphertext and execute another bitwise exclusive-OR; the output will be the plaintext.

To see how this works, let's walk through a simple example. In the computer, step 1 is done for us already. The only way the computer can handle alphabetics is by representing them as bit strings. Fig. 1 shows the encryption of the message, "Hello," by Vernam's method. The first bit string is the ASCII representation of the message; the second bit string is the key—a random sequence of 1's and 0's.

To see how we arrive at the third bit string, you must look briefly at the exclusive-OR operation, shown in the table in Fig. 2. The usual way of describing this is to say that p XOR q has a value of 1 if either p or q is 1, but not both.

But there is another, more meaningful, way of looking at this. Look at Fig. 2 again and notice that if q = 1, the operation complements the p bit, while if q = 0, the p bit is left unchanged. XOR can therefore be viewed as a sort of controllable bit-flipper, and this is the way it functions in Vernam's algorithm.

Returning to Fig. 1, you will see that at every place where the key bit is 0, the message bit is copied into the cipherbit unchanged, and where the key bit is 1, the cipherbit is the complement of the message bit. Vernam's method thus flips bits in the message at random. To decipher the message, we apply the same key; this time it flips these same bits back again and restores the plaintext.

This system is a good deal more powerful than that description might lead you to expect. The key to the system, and to all its strengths and weaknesses, lies in the word "random." What makes a key random, anyway? Without going into all of the philosophical ramifications of probability theory, we can say that for our purposes a bit string is "random" if it is unpredictable and "almost random" if it is extremely difficult to predict.

An almost-random key is usually generated by some controllable process, and the whole trick of cracking a cipher consists of unearthing the controllable process and thus making the key predictable. A truly random key is generated by a naturally unpredictable process, like the tossing of a coin or the random emissions of radioactive decay. A Vernam cipher using a truly random key, and using it only once, is absolutely uncrackable; the reasons for this are elegantly explained in Kahn's book. (Stealing a copy of the key will enable you to read the message, but theft is not ordinarily considered a cryptanalytic technique.)

The key in my program is almost random. The commonest way of getting an almost-random key is by deriving it from a keyword. The main problem with this is that any keyword will be too short. That's because any almost-random key will eventually have to repeat, and discovering and analyzing these repetitions is the primary method of attack in cryptanalysis. The longer the period of the key, the harder it will be to mount such an attack.

The easiest way to generate a long-period key was discovered by a colleague of Vernam's. He realized that if you used two keys of different lengths, the pattern of the combined key would not repeat until the two words got back in step again.

For example, in Fig. 3 I use the keys BOY and GIRL. You will see that, for example, B and G occur together the first time, and then get out of step, until after four repetitions of BOY and three repetitions of GIRL they are back in step again. This illustrates the general rule: if keywords are used whose lengths have no common factors, the period of the composite will be equal to the product of the lengths of the keywords.

My program uses three keywords (and could quite easily be extended to more), and is capable of quite long periods. For example, the keys PHILANTHROPIC, BITTERSWEET and HENDECASYLLABIC when combined will yield a composite key with a period of (13*11*15) = 2145 characters.

The program is given in Listing 1. Lines

Listing 1. File encryption program in BASIC.

```
100 REM
110 REM ************* FILE ENCRYPTION PROGRAM *************
120 REM
130 REM    USES VERNAM'S ALGORITHM WITH THREE KEYWORDS
140 REM
150 REM    ALAN SCLAWY, JULY, 1980
160 REM
170 REM ***************************************************
180 REM
190 REM VARIABLES:
200 REM
210 REM   F$              FILE NAME
220 REM   K1$,K2$,K3$     KEYWORDS
230 REM   K1, K2, K3      POINTERS TO CHARACTERS IN KEYWORDS
240 REM   L1, L2, L3      LENGTHS OF KEYWORDS
250 REM   IN$             TEXT LINE CURRENTLY BEING ENCRYPTED
260 REM   EN              LENGTH OF IN$
270 REM   Z$              CHARACTER FROM IN$ CURRENTLY BEING
280 REM                     ENCIPHERED
290 REM   P               HOLDS NUMERICAL EQUIVALENT OF Z$
300 REM   Y$              ENCRYPTED VERSION OF Z$
310 REM   OT$             ENCRYPTED VERSION OF IN$
320 REM
330 REM ***************************************************
340 REM
350 REM ------- MAKE ROOM FOR STRINGS
360 CLEAR 400
370 REM
380 REM ------- GET FILE NAMES & OPEN FILES
390 INPUT "Input file"; F$
400 OPEN "I", 1, F$
410 INPUT "Output file"; F$
420 OPEN "O", 2, F$
430 REM
440 REM ------- GET KEYWORDS
450 INPUT "Key"; K1$
460 K1 = 1
470 L1 = LEN(K1$)
480 INPUT "Key"; K2$
490 K2 = 1
500 L2 = LEN(K2$)
510 INPUT "Key"; K3$
520 K3 = 1
530 L3 = LEN(K3$)
540 REM
550 REM --------- LOOP ON RECORDS ------------------------
560 FOR I = 1 TO 32767
570 IF (EOF(1)) THEN 920
580 LINE INPUT #1, IN$
590 EN = LEN(IN$)
600 REM PRINT "EN ="; EN
610 REM PRINT IN$
620 OT$ = ""
630 REM
640 REM --------- LOOP ON CHARACTERS WITHIN RECORD --------
645 IF (EN=0) THEN 880
```

```
650 FOR J = 1 TO EN
660 Z$ = MID$(IN$, J, 1)
670 U$ = MID$(K1$, K1, 1)
680 P = ASC(Z$) XOR ASC(U$)
690 K1 = K1 + 1
700 IF (K1 > L1) THEN K1 = 1
710 V$ = MID$(K2$, K2, 1)
720 P = P XOR ASC(V$)
730 K2 = K2 + 1
740 IF (K2 > L2) THEN K2 = 1
750 W$ = MID$(K3$, K3, 1)
760 P = P XOR ASC(W$)
770 K3 = K3 + 1
780 IF (K3 > L3) THEN K3 = 1
790 Y$ = CHR$(P)
800 REM --- PREVENT INADVERTENT NULL, CR, LF, OR EOF
810 IF P <> 0 AND P <> 10 AND P <> 13 AND P <> 26 THEN 830
820 Y$ = Z$
830 OT$ = OT$ + Y$
840 REM PRINT J; ASC(Z$), U$; V$; W$; ASC(Y$)
850 NEXT J
860 REM --------- END OF LOOP ON CHARACTERS ---------------
870 REM ------- WRITE OUTPUT RECORD
880 PRINT #2, OT$
900 NEXT I
910 REM --------- END OF LOOP ON RECORDS -----------------
920 CLOSE 1, 2
930 END
```

```
(a)  Encryption

     RUN
     Input file? PLNTXT
     Output file? CYPTXT
     Key? ALPHA
     Key? BETA
     Key? EPSILON
     OK


(b)  Decryption

     RUN
     Input file? CYPTXT
     Output file? CLRTXT
     Key? ALPHA
     Key? BETA
     Key? EPSILON
     OK

(c)  Input message

          This is an example of a paragraph of text encryp-
 ted by means of Vernam's algorithm.  In encryption, bits in
 the ASCII codes for the characters are flipped at random in
 a pattern set by the key, transforming the message into
 gibberish.  In decryption, the same bits are flipped back
 and the original text is restored.  If the key is per-
 fectly random, the cipher is uncrackable; if the key has a
 long period, the cipher is crackable only with difficulty.

(d)  Output ciphertext

 OT?)<k?'z62a8;1=-05o"*t;j86-.34%.*{*8f!)43a<>.&'41
   1;?a87k(*(!5j#5w87$10f7y.,7#!#1#*htk(k>&!,%<526>|`*/92h+-
 <2&wCr&<>(61,23y91'|+-2580:#+$`.93t<;51-&4p<(p=,"05'h>1
 .t6%*6>70f&)8g# p9<;d* 'wa.<*+</ 4'%=0q)-/p0$7*.'51:$1$
  /6)=4"( 1~|/{=532169(',oh.+2}5(/?r':.>e-88a?!02,-!s%80%
  '73`;#3t5%5&4-1<}(5791=)j:2,;;4!:1{e u8$"a254t77a5;)1
 <+(1#0o4+"78<qe>88a'0?(5>s#6k2(799% :*.;g1(=y$8%h-(8h*";z"
 ;2(.b*77:5)i1>5$y.024-7s.*s-484+.):1z82-$c'9)4p+$*23)=;+6z
```

Fig. 4. Sample program run with input and output.

390 to 420 identify and open the file being encrypted and the file that receives the encrypted version. Lines 450–530 accept keywords from the user. The program recognizes the difference between upper and lowercase keywords.

Lines 560–900 are a big loop for reading, enciphering and writing lines from the source file. Input is done by a line input statement in line 580. We need to use line input because an ordinary input statement will read a character string up to the first comma and then think its job is done. You want to handle bigger chunks of data than that. The line input statement will read characters until it finds a carriage return (CR). Ideally we would like to read a whole sector from the disk each time, but that requires I/O capabilities that are not provided in some dialects of BASIC; using line input is a compromise.

Lines 650–850 are a loop for enciphering one line of text. Each character is selected from the line, using a MID$ operation, in line 660. Vernam's XOR operations can be seen in lines 680, 720 and 760. (BASIC will not let us do XOR operations between characters, so you have to convert them to numbers by means of the ASC function and then convert the result back again by means of the CHR$ function at the end.) First you XOR the plaintext with key #1 in line 680; then you XOR the result with key #2 in line 720; and finally you XOR *that* result with key #3 in line 760. Key letters are selected each time by MID$ operations, and after each XOR the key's index is advanced and wrapped back around to the beginning of the key if necessary.

The I/O conventions of CP/M and BASIC pose some special problems for us. First, a control-Z character is used by CP/M to mark the end of a file. Therefore, if you should accidentally hit upon a combination of message and key characters that together produce a control-Z, this will be written onto the ciphertext file. When you decipher, the program will find this control-Z and think that the file ends there, and all the rest of the file will be irrecoverable. Similarly, an accidentally produced CR will be treated as a delimiter by the line input operation.

I also found out, the hard way, that BASIC will not insert a null into a character stream, and as a precaution I thought it wise not to allow line feeds. Thus whenever any of these characters results from the encryption process, you give up and retain the plaintext character. I feel a little uncomfortable about retaining plaintext characters, but since these situations arise infrequently and at random, it is probably safe to do so. Lines 810 and 820 test for these cases and take care of them. Each new ciphertext character is appended to the output string in line 830, and in line 870, after the line is

complete, the output string is written into the output file.

The program includes a few added print statements in lines 600, 610, 840 and 890 for test and verification purposes. For actual use, these statements are disabled by turning them into remarks, but I have left them in for your convenience. The program generally is written a little less tightly than it might have been; I did this for ease of development and left it that way for clarity.

Fig. 4 shows a typical encryption run followed by a typical decryption run, together with the plaintext used and ciphertext obtained. (The correspondence between the two texts is not exact, since some of the ciphertext characters are non-printing.) Using a 2 MHz CPU and Microsoft's Version 4.3 BASIC under CP/M, it takes just about a minute to encipher the sample paragraph shown in Fig. 4. Execution can be speeded up by deleting the remarks and combining some of the statements. For real speed, however, you would be better off using a compiler-type BASIC, or hand-compiling the program in assembler language.

A word about keyword selection. Ideally a keyword should be a completely random bunch of symbols, and it should be easily memorized so it does not have to be written down. Of course, these are contradictory requirements. The important points to remember are these:

First, keep the keywords long, so the composite key period will be as long as possible. Ideally, the period of the key should be at least as long as the file you're encrypting.

Second, don't use keywords that have anything to do either with yourself or with the matter that you are encrypting. Using your own name or your spouse's name is *out!* The temptation to draw keywords from the file itself is nearly irresistible, but you must resist it. When encrypting a file containing notes for a patent application, don't key it with words like "invention," "concept" or "disclosure." Use crazy words like "gesundheit," "anamorphosis" or "3nB%r-7*Q9Xm<."

Third, remember that the lengths of the three keywords must have no common factors if the composite period is to be as long as possible. For example, Fig. 3b shows what happens if the keywords both have a factor of 2. The individual keys have lengths of 4 and 6, but you will notice that the composite repeats after 12 characters instead of 24.

Choosing three long keywords of mutually prime lengths all unrelated to the material you are protecting may seem tedious and fussy, but armed with such a key, this simple encryption program will give even a professional codebreaker a fair amount of trouble.■

Electrocardiogram For Your Computer

This alarming device monitors your computer's pulse rate and signals in the event of cardiac (or CPU) arrest.

Kenneth H. Reid
1935 Trevilian Way
Louisville, KY 40205

When a microcomputer is used in a dedicated real-world control application, the consequences of system failure may be serious. Any system whose failure would possibly produce injury, equipment damage, crop failure or other financial loss should have a simple, foolproof alarm device that provides a positive indication that the system is working when it is and an equally positive and compelling alarm signal when the system fails. A single chip circuit can perform these tasks.

The chip is the 4049, a CMOS hex inverter. The principle of the alarm circuit is to monitor a pulse emitted by the microcomputer; if the pulse fails to recur on schedule, the alarm is activated.

One inverter is used to sample and hold a positive transition; this, in turn, gates two oscillators. Each oscillator uses two inverters, for a total of five. The sixth inverter is available as an alarm output buffer.

The Circuit

The circuit is illustrated in Fig. 1. In my application, the alarm is located about 1000 feet from the computer, beside a remote terminal. The computer operates a polling loop, cyclically sensing data from 24 input stations and responding to it when it appears.

Once per cycle (about two seconds) the computer sends a single nonprinting character to the remote terminal. This is received as an indication of proper function. When data is received, printing characters are sent, and these also are accepted as OK signals by the alarm circuit. If for any reason the program leaves the polling loop and comes to a halt, the train of signals stops and the alarm is activated.

The heart of the circuit is the sample-and-hold, shown in the center part of Fig. 1. The first RC circuit merely blocks out dc on the signal line. The two diodes act as a half-wave rectifier, charging the capacitor whenever the signal goes positive.

The RC time constant of the hold circuit can be set to suit the signal; a good 0.5 uF capacitor with no resistor has a time constant of 10–20 minutes in a dry environment. The resistor is recommended, if only to reduce sensitivity to local humidity.

Heart Beats

The safe signal needs to be specified with some care if it is to be monitored all day long in a general office or home environment. Its purpose is to reassure. The safe oscillator, by flashing a green LED, confirms that the system is operating. By flashing at the right frequency, it conveys the message "everything is OK." The right frequency is 40–60 flashes/minutes, the heart rate of a relaxed person.

The trouble signal must convey the opposite message. The oscillator frequency chosen for this signal is 200–240 flashes/minute, the heart rate of someone thoroughly alarmed. In addition to the red LED, a high-frequency solid-state beeper sound generator is driven at this same rate, providing a sound signal whose urgency is almost impossible to ignore.

Once the emergency is recognized, the sound signal can be turned off by a switch, but the red LED continues to flash until the trouble has been corrected.

Obviously, the alarm circuit should not be disabled by a power failure. I use a rechargeable 6 volt battery, maintained by a simple trickle charger. With a total current drain under 2 mA, the battery can power the alarm for a month. A standard lantern battery should last six to eight weeks.

I have used this alarm now for several months. During this period it has detected four program dropouts (illegal jumps) that were probably caused by power line transients and three disk reading errors. These were all soft errors, easily corrected, which nevertheless completely immobilized the system. ■

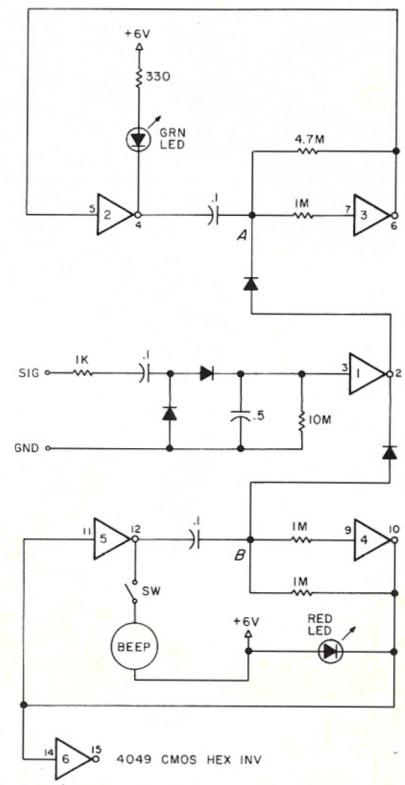
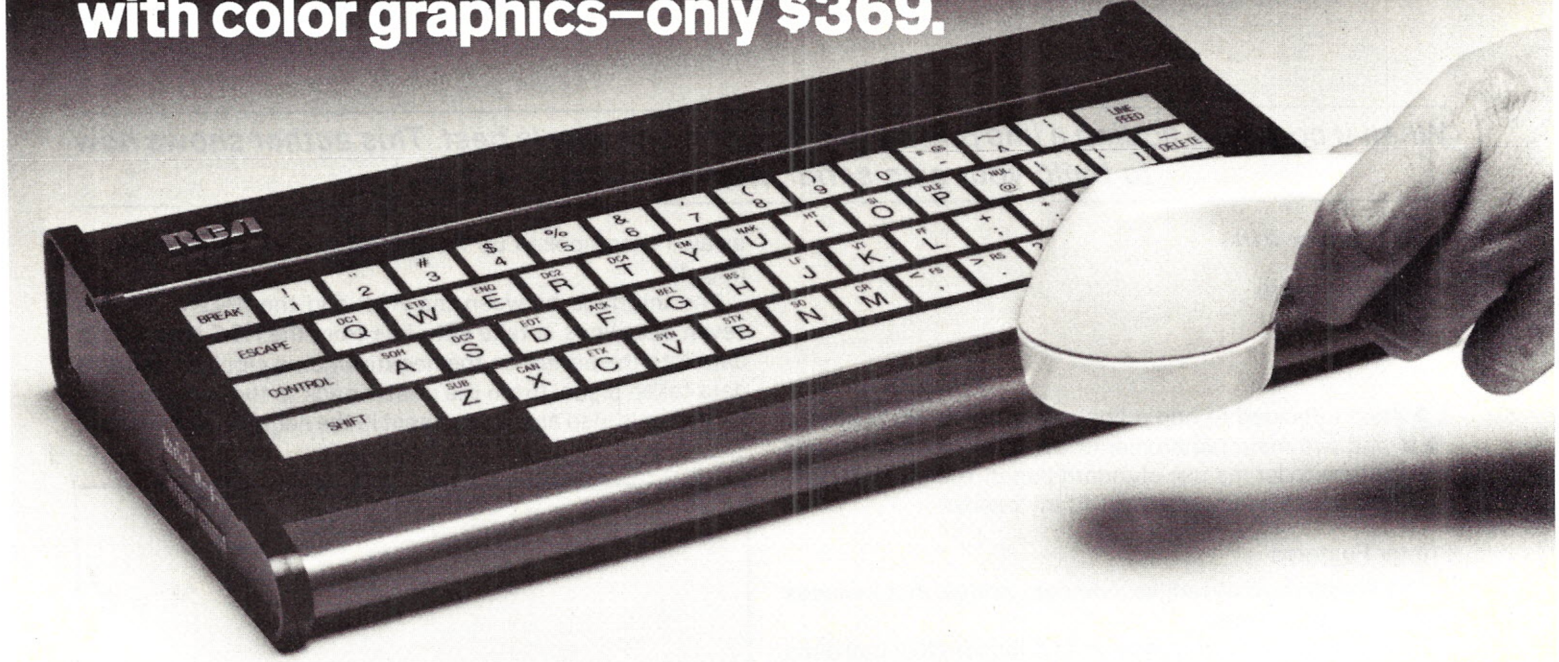


Fig. 1. The alarm circuit consists of a safe oscillator (top), a signal detector (center) and a trouble oscillator (bottom). When signals occur on the RS-232 input line, low-to-high transitions are captured and hold the input of inverter 1 high. The output goes low, stopping the trouble oscillator by pulling point B low and releasing the safe oscillator. If the incoming pulses stop, after a period determined by RC, the input of inverter 1 goes low. This causes the output to go high, locking point A high and releasing the trouble oscillator, causing the alarm to sound. The output of inverter 6 will be high in the safe condition and will be an intermittent current sink in the trouble condition.

Suddenly, RCA makes talking to your computer a lot cheaper.

New interactive data terminal with color graphics—only $369.*



RCA's new VP-3301 is a professional quality, ASCII encoded, interactive data terminal, suitable for a wide variety of industrial, educational, business and individual applications requiring interactive communication between computer and user. Connects directly to your computer or to a standard modem for over the phone access to time sharing networks and data bases. And it's compatible with networks such as those provided by CompuServe Information Services and Source Telecomputing Corp. Microprocessor intelligence and LSI video control integrated circuits bring performance, features and flexibility at a low price. Operates from 5 volt power supply (included).

Unique color locking circuitry creates sharp, jitter free, true color graphics and rainbow free characters.

You can display the entire field of characters in any of 8 colors against any of 8 background colors (8 gray scales with monochrome monitors). Or to add special emphasis, you can display individual letters, words or lines in different colors or in reverse video.

The VP-3301 offers you a choice of two software-selectable display formats: Either 40 characters by 24 lines. Or 20 characters by 12 lines.

The terminal's resident character set consists of 52 upper and lower case alphabetics, 10 numerals, 32 punctuation/math symbols and 31 control characters.

You can also define a total of 125 of your own characters. Including: Greek letters and other foreign alphabets, graphic symbols, large graphics building blocks, playing card suits, unique character fonts and "little green men."

The keyboard section features flexible-membrane key switches with contact life rated at greater than five million operations. A finger positioning overlay and positive keypress action give good operator "feel".

An on-board sound generator and speaker provides aural feedback for key presses and may also be activated with escape sequences to provide an audio output.

The sealed keyboard surface is spill proof and dust proof. This, combined with high noise immunity CMOS circuitry, makes the VP-3301 ideal for hostile environments.

Output is industry standard asynchronous RS232C or 20 mA current loop with 6 switch-selectable baud rates and 8 selectable data formats.

You can connect the terminal directly to a 525 line color or monochrome monitor. Or to a standard TV set using your RF modulator.

For more information, contact RCA MicroComputer Marketing, New Holland Avenue, Lancaster, PA.

Or call our toll-free number: 800-233-0094.

*Suggested user price.

RCA

✓30

Printer Interface for the H8 (I)

When it comes to adding a line printer to the H8, no one way is best. This author shows how to build your own serial interface for the IP-225...

Norman S. Dick W1NS
4 Fullin Court
Norwalk, CT 06881

When I decided to add a line printer to my H8 system, I had two main requirements: an 80-column format and impact type to let me use standard paper. The Integral Data Systems IP-225 seemed to best fit my needs.

Printer Features

The IP-225 is an attractive, compact printer that includes the following features:

- 17.25 inches (43.8 cm) wide × 11.5 inches (29.2 cm) deep × 7.0 inches (17.8 cm) high
- Serial RS-232C interface or parallel TTL-level interface
- Full lower and uppercase ASCII character set
- Microprocessor controller
- 7 × 7 dot matrix
- 8½ inch wide paper—roll, fanfold or sheet
- Serial baud rate to 1200 bits per second
- Sustained throughput to 50 characters per second
- Line buffer of 256 characters
- Built-in-test mode, in which the printer will output a continuous full line repeating alphanumeric pattern
- Form-feed control

A more inexpensive version (IP-125) is available with friction instead of tractor feed and does not include the automatic form-feed option. But this feature is useful during extended printouts. Each page has a nice margin at the bottom, which makes printout duplication easier.

The printer is fairly quiet and generates superb quality printouts. The built-in-test feature is useful for isolating system problems.

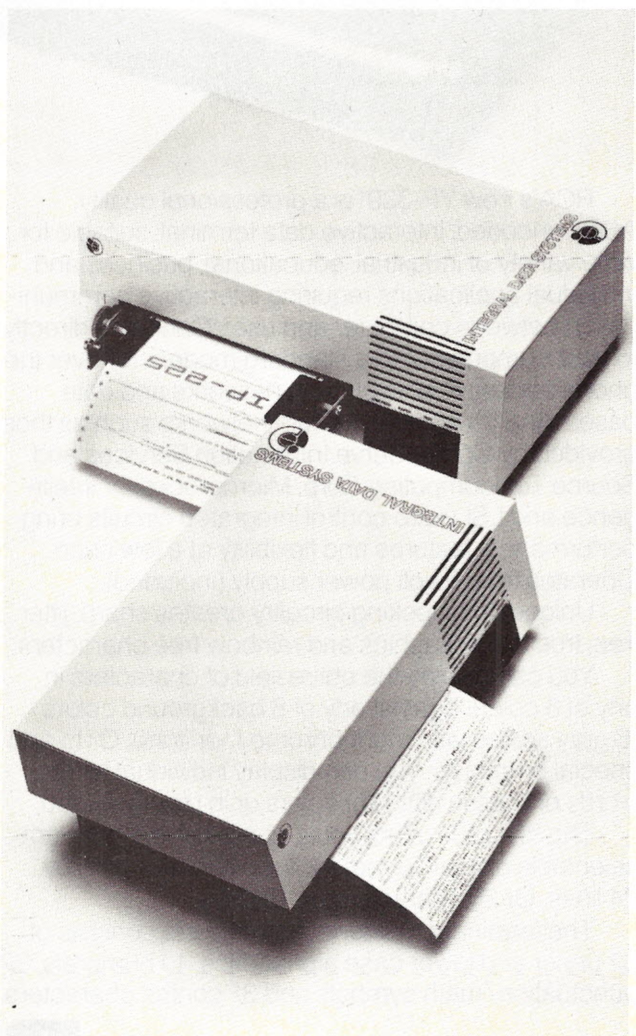
Unfortunately, the 80-column output is only 77 columns. The print density is ten characters per inch, and the tractor-feed holes limit the useful width to less than eight inches. (This is not a problem in the IP-125.)

Some people may object to the upside-down printouts, with respect to the front of the printer, but I don't find this annoying.

Operation

When my IP-225 arrived, I loaded some paper, placed the printer in the test mode, and it immediately began chugging away. I then hooked up the printer to my parallel interface (Fig. 1).

The interface between the H8-2 and the IP-225 includes seven data lines, a take data line from the H8-2 that clocks data into the printer on the negative-going clock edge and a data taken signal (from the printer back to the parallel interface) that is also asserted or sent in the negative-going state.



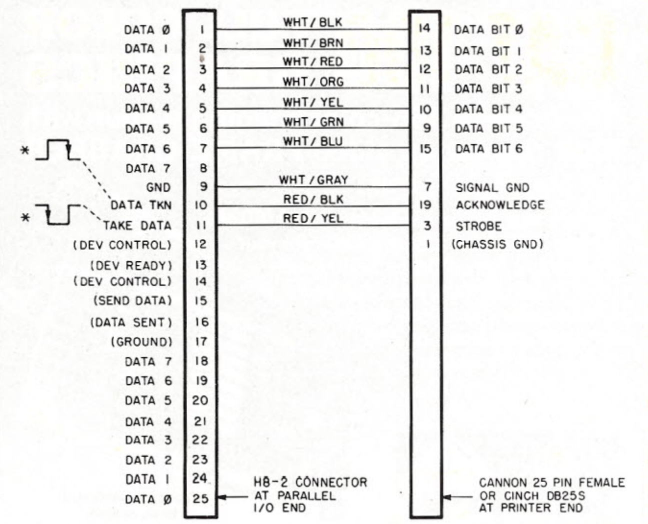


Fig. 1. Printer to parallel interface cabling. Initial hookup.

DATA TKN is activated by negative edge.
Data should be clocked into printer on leading (negative) edge of TAKE DATA.
In the IP-225 printer, cut etch jumper Z2 to Z4 and connect Z1 to Z3 (configures handshaking polarities).
In H8-2, configure jumpers as follows: A1-A2 open (noncontinuous mode), E1-E2 shorted (true data), F, G, H open (interrupts).

Fig. 1 notes.

The IP-225 uses the handshaking method of interlocked communication. At each end of the interface cable, each device will send its control signal and await a response before resending. The parallel I/O sends TAKE DATA, the printer responds by sending DATA TAKEN, and the parallel I/O, upon recognizing DATA TAKEN, may output the next character on the data lines by sending the take data strobe.

It took a while to dig out of the Heath manuals the information I needed to initialize the USART on the parallel interface and manually output characters to the printer from the H8 front panel. Upon receipt of a carriage return character by the printer, preceding characters that enter the printer line buffer are printed out.

After successfully printing characters via the H8's front panel, I then tried outputting to the printer from a BASIC program, by using the PORT command to switch the output port over to the parallel I/O instead of to the H9 terminal. But the printer either hung up dead or slewed over to one side and started grinding away. Nothing would clear the grinding except removing and reapplying power to the printer.

Printer Connection

I finally discovered that the handshaking was not being obeyed by the parallel interface. The H8-2 was sending over characters without waiting for an acknowledge from the printer. The take data strobe was causing an interrupt to the internal microprocessor within the printer. If this interrupt occurred before the previous character was completely processed, the printer's firmware crashed.

Discussions with Heathkit confirmed the problem. When using Heathkit's BASIC, the H8-2 parallel interface looks software-identical to the H8-5 serial interface. Characters are spewed out at a high rate from the parallel interface without checking the status of the acknowledge from the printer. Hence, the handshaking is lost.

To successfully use the H8-2 with the IP-225, or any other printer with parallel interface, you must write a software subroutine to allow monitoring of handshaking signals from the parallel interface. Since Heathkit did not provide a source listing of their BASIC V10.02.01, it was virtually impossible to incorporate such a subroutine to the existing BASIC. (Note: Heath has since developed later BASIC versions for use with their own H-14 line printer, in conjunction with a serial interface (H8-4) where handshaking is provided by a bus 1 control signal.)

At this point I stared in disgust at my expensive line printer and H8-2, wondering if I would ever get them to talk to each other.

I then abandoned the H8-2 and started thinking of the serial mode of the printer. The H8-5 serial interface could be used with the IP-225 at 1200 baud, but only if you use the clear to send line in the serial interface. If the printer buffer is full, new data should not be sent by the H8-5, since the printer buffer will be overwritten and data already in the line buffer will be lost.

However, using ten characters per inch, the printer is fast enough to receive at 300 baud without the data buffer being overwritten. Therefore, the printer can be used as a slave to the H9 terminal merely by setting up the H8-5 and H9 terminal for 300 baud and wiring the serial input of the printer in parallel with the serial input to the H9. You need no additional interface cards. (Caution: other line printers may excessively load the serial input to the H9 depending on receiver design.)

This slaving technique is applicable to many other systems besides the Heathkit. Indeed, several higher-priced terminals provide a printer output connection for use in the slave mode. The slave technique is useful primarily for hardcopy printouts of program listings. The slave mode's disadvantage is that information cannot be output to the printer and to the terminal independently, under software control.

Fig. 2 summarizes all switch settings and connections that must be made to use the IP-225 as a slave to the H9. I'm using a 20-foot twisted pair cable from the back of the H8 computer to the input of the IP-225 with no problems. When power is turned off in the IP-225, the H9 terminal still functions normally, since the IP-225's receivers do not substantially load the RS-232 interface lines with power removed.

The printer is now a pleasure to use, with no additional hardware interfacing or software subroutines required. I recommend ordering the IP-225 with the parallel I/O feature since this is a free option and switches are provided internally to revert to the serial mode of operation.

The moral of the interfacing story is: Simplest is Best! ■

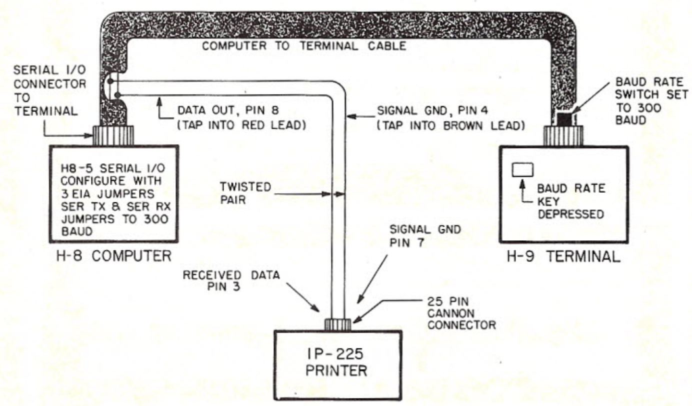


Fig. 2. Final system connections.

✓73

Printer Interface for the H8 (II)

. . . Or, you can go the parallel route, as this user did with his IP-125.

Howard L. Cunningham
330 Blumen Lane
Dayton, OH 45418

Soon after my H8 system was operating, I needed a printer. This need was reaffirmed each time I had to record a listing by hand. After perusing the Heathkit catalog to determine the requirements and cost of the H14 printer (both kit and assembled), I decided to look elsewhere.

I had always wanted to design and build something for my own use, so I decided to buy a printer and design the parallel interface, if necessary. I selected the IP-125 from Integral Data Systems. An associate had already purchased one that I felt was cost effective.

The IP-125 is a dot matrix plain paper printer with a TTL-parallel or RS-232C serial interface standard (switch selectable). With a common option, you can vary the print widths. After reviewing the H8-2 parallel interface design, which uses back-to-back UARTs to implement a parallel interface, I decided that the Intel 8255A PIA should serve as a basis for my design.

For my software design, I decided to implement the printer software driver as a patch to the Heath CRT driver, which is common to all Heath software. I planned to intercept all characters going to the CRT and echo them optionally on the printer or CRT, or both. This technique allowed me to use existing Heath software and obtain hard copy from all Heath software.

Photo 1. Author's H8 computer system.

Photo 1 shows my system, which includes H8 with 16K RAM, dual 1200 baud audio cassettes with homemade cassette controller, H9 CRT and IP-125 printer. Note that the printer control switch is in the small box between the H8 and H9. The Gould OS245A dual-trace oscilloscope is also pictured.

Hardware

The interface schematic is shown in Fig. 1; Table 1 shows the parts list. The critical element on the parts list is the 8255A, which has improved timing characteristics over the 8255, so don't purchase the wrong chip. The 8255A costs about $9 from any Intel distributor.

Memory-mapped I/O is used for this interface. This was forced during board debugging. After some problems, I discovered that the Heath serial I/O board (H8-5) appeared to always have its bus transceivers on for any I/OR or I/OW operation. The Heath schematic confirmed this, so I decided to let Heath keep the I/O instructions and use memory instructions.

U4 and U5 serve as address bus buffers. Address decoding is accomplished by U6 and U7. Board operation lacks exhaustive decoding; that is, the board will not only respond to addresses 200.000, 200.001, 200.002 and 200.003, but will also answer 2xx.00 – or 3xx.00 –, where x is any digit (0–7) and – represents 0, 1, 2 or 3.

This technique saves decoding logic. When the computer executes a memory instruction to the proper address, U6 produces the board select signal, which is $\overline{\text{CS}}$ for the 8255A. This signal also gates MEMW and MEMR to the data bus transceivers to complete a data path to or from the 8255A. The Intel data sheet describes the PIA operation.

The low-order seven bits of port A comprise the printer data. The data is sent through the line drivers U9 and U10 to the printer. In the strobed output mode, bit 6 of port C goes low ($\overline{\text{OBF}}$) whenever data is written to the PIA port A. This signal is passed to the IP-125 as its strobe ($\overline{\text{STB}}$). When the IP-125 is ready for more data, it lowers its acknowledge line ($\overline{\text{ACK}}$), which goes into the PIA on port C, bit 7, to clear $\overline{\text{OFB}}$ to high, completing the handshaking. $\overline{\text{ACK}}$ also goes in-

to port C, bit 2, for use by the driver.

The printer control switch is brought into port C, bits 4 and 5. Port C, bits 0, 1, 2, 3, 4 and 5, are set up as input mode 0 (no handshaking). Port B and bits 0, 1 and 3 of port C are not used in this interface. See the completed board in Photo 2.

Software

The driver source listing places a JMP LPDRV at 040.363 of the Heath CRT driver. At LPDRV, the character to be printed is in the A register and on the stack (via a PUSH PSW). LPDRV first programs the PIA by storing octal 251 in port C (address 200.003). This sets port A in mode 1 (strobed) output, bits 4 and 5 of port C as input, port B in mode 0 as output and bits 0, 1, 2 and 3 of port C as input.

The driver then checks the printer control switch. If bit 5 of port C is off (CRT ONLY switch position), the driver jumps to ENDPRT to avoid printing. If bit 5 is on (PRINTER ONLY or BOTH switch positions), the driver reads port C to check if $\overline{\text{OBF}}$ and $\overline{\text{ACK}}$ are high. If either line is low, the driver spins until both are high. The driver then executes POP PSW and prints the character at port A (address 200.000).

After printing, the driver again waits for $\overline{\text{OBF}}$ and $\overline{\text{ACK}}$ to go high. This dual-spin loop check ensures that no data is ever sent to the printer when $\overline{\text{ACK}}$ is low. If this is done, the IP-125 can jam the printing head necessitating a power on-off cycle to clear the malfunction.

After printing, the driver executes PUSH PSW and falls to ENDPRT. Here the driver checks the printer control switch (bit 4, port C). If bit 4 is low (PRINTER ONLY switch position), the driver jumps to ENDCRT. If bit 4 is high (CRT ONLY or BOTH switch positions), the driver executes POP PSW and JMP $CDOUT. $CDOUT outputs the character to the CRT and returns to the caller. ENDCRT executes POP PSW and returns to the caller.

Note the six assembly errors in the listing. Although the assembler (HASL #04.01.01) generated the correct code, all references to 16 operands with bit 16 on generated a "V" error, even though the instruction appears correct. This inconsistency in the assembler could occur in a computer whose maximum memory is 32K (177.377) and memory-mapped I/O is not expected.

Printer Installation

In order to use the printer driver with existing Heath software (TED, HBUG, HASL, B.H. BASIC and Extended B.H. BASIC), the software must be configured to limit high memory below the driver. For my 16K system, I configured high memory to 24,437, or 137.165, leaving 140 bytes for the driver. Although the driver only needs 52 bytes, I allowed for space above the driver for the PAM stack area.

While all cassette-loaded Heath software will set the stack pointer to high memory-1, PAM will set the stack pointer at the physical top of memory whenever you press RST/0. If you place the driver too close to the physical top of memory, the

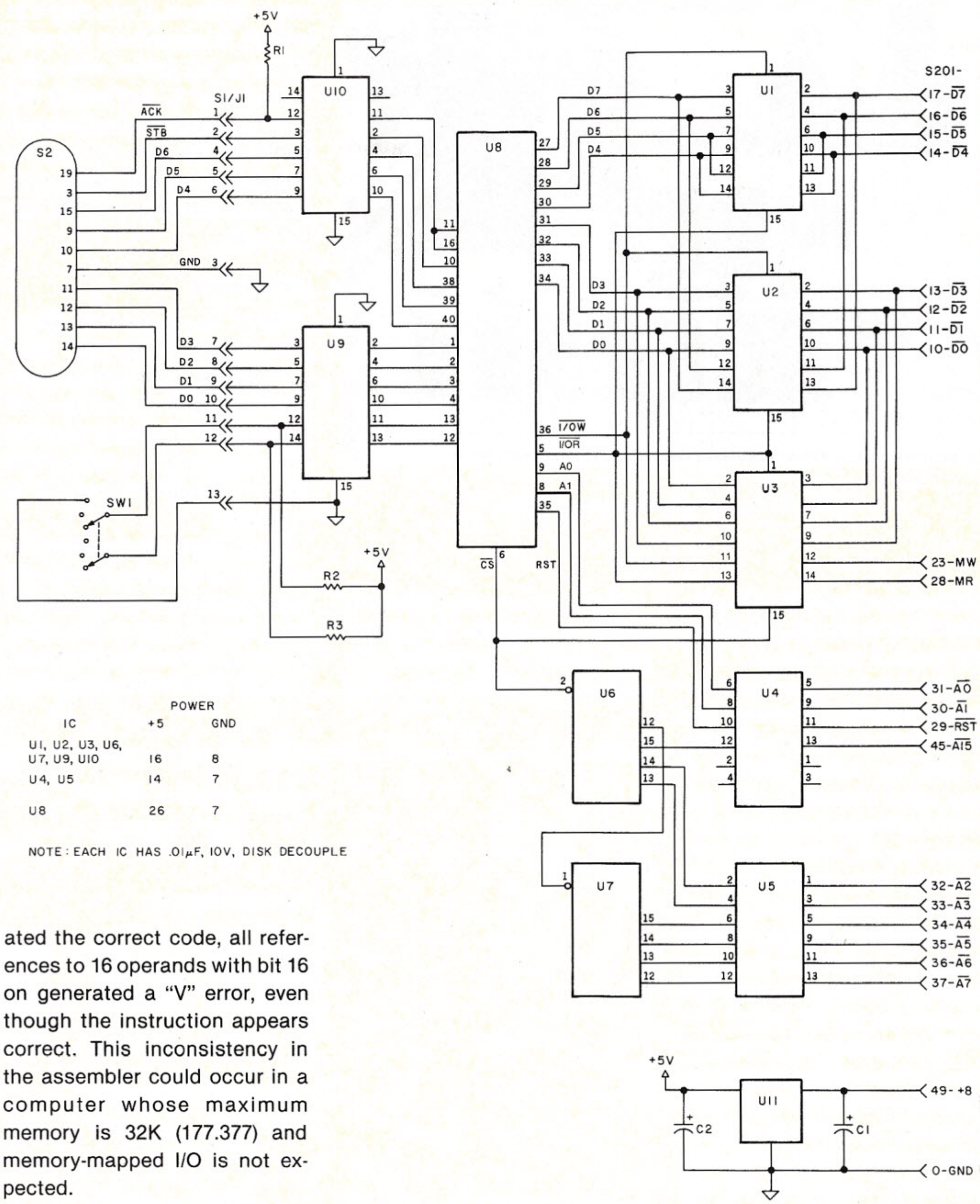


Fig. 1. Interface hardware schematic.

Integrated Circuits

U1,U2,U3	74LS368
U4,U5	74LS04
U6,U7	74LS42
U8	P8255A
U9,U10	74LS367
U11	7805

Discrete Components

C1	10 μF, 15 V, electrolytic
C2	33 μF, 16 V, tantalum
C3–C12	.01 μF, 10 V, disk
R1,R2,R3	4,7k Ω, ¼W or other TTL pull-up resistor
SW1	DPDT center off rocker switch

Connectors

S201	2 Heath circuit board sockets (432-947)
S1	1 Heath 25-hole connector shell (432-938)
J1	1 Heath 25-pin plug (432-946)
S2	DB-25S

Table 1. Parts list.

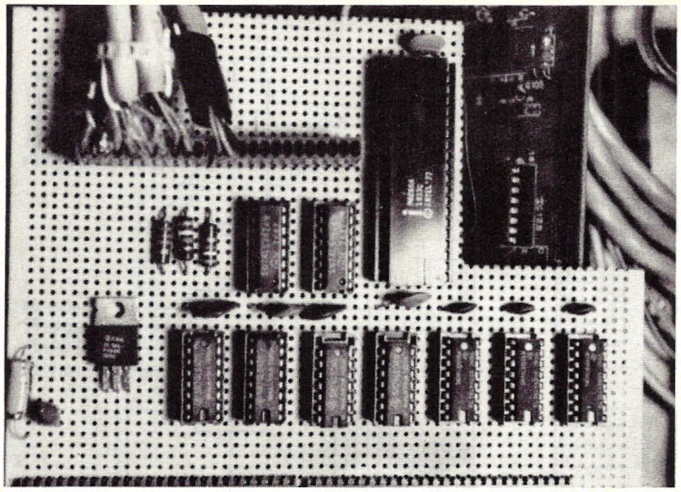


Photo 2. Printer interface circuit board.

PAM stack will wipe out the driver if RST/0 is pressed. RTM/0 will not cause a stack reset.

In the interest of speed, I recommend that you load the driver separately, rather than integrate it with any Heath software. If you dump the driver with any Heath software to an audio cassette, you will dump most of the memory (040.100 to 137.251). This will take longer to load. If loaded separately, the driver will load in two segments—one will patch at 040.363, and the second will load at 137.165.

You should load the driver before pressing GO after loading the Heath software. If you forget to do this, press RTM/0, REG PC and remember the PC. Then load the driver, reset the PC and press GO to return to the Heath software.

Operation

Let's consider a typical system start-up sequence. After powering up, load the desired Heath software tape. Now load the driver tape before pressing GO. Check the printer control switch.

If you want complete hard copy, set the switch to BOTH. If not, set the switch to CRT ONLY. Switching to PRINTER ONLY will prevent commands from echoing on the CRT. Press GO.

If you ever list to the CRT ONLY and want hard copy, simply press PRINTER ONLY or BOTH. If you are listing in BOTH mode and wish to print faster, go to PRINTER ONLY.

The CRT is character-level buffered, so it can be stopped and restarted in midline with predictable results. The printer is line buffered. If you stop it in midline, it remembers the characters in the buffer. When you then restart it, the printer will print the old buffer data before it prints the new data. To clear the printer buffer in this case, cycle the power off-on in the CRT ONLY mode.

Conclusions

Don't be concerned about your printer sitting idle while you work on the interface modification. The IP-125 can run in a serial mode at greater than 300 baud. Thus, you can daisy-chain the IP-125 right off your H9 at 300 baud.

First, strip a small piece of insulation off the ground wire and the received data wire at the H9. Next, run two jumper wires from the data line on the H9 to the data pin (pin #3 on the DB-25) on the IP-125 and from the ground on the H9 to the signal ground (pin #7 on the DB-25) on the IP-125. The H9 must run at 300 baud, since this technique ignores the clear-to-send signal from the printer. If the H9 is run faster, you will send data when the IP-125 cannot handle it.

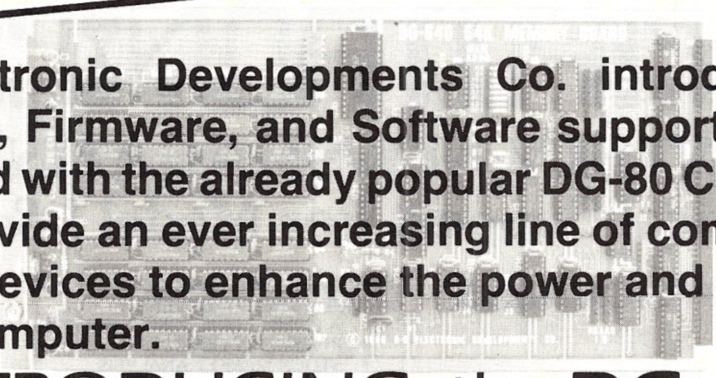
To run your H9 at two baud rates (300 and 4800, for example), you can add a DPDT switch to the H8-5 serial board to vary the H8 transmission rates, just as the H9 supports two rates via a switch. Now you can run CRT dialog at high speed with the IP-125 powered off. If you want hard copy, switch to 300 baud (H8 and H9 both), switch the IP-125 on and run at 300 baud.

There you have it: a parallel interface for a printer and a way to get operating without it. ■

```
                               * PATCH CONSOLE DRIVER
                               *
    040.363                          ORG     40363A
    040.363   303 165 137            JMP     LPDRV          INTERCEPT OUTPUT TO CRT FOR PR
INTER
                               *
                               * DRIVER EQUATES
                               *
    040.100                    START  EQU    40100A         PC START ADDRESS
    040.111                    $CDOUT EQU    40111A         OUTPUT CRT LOW LEVEL
                               *PICTL EQU    200003A        PIA CONTROL WORD
                               *PRCTL EQU    200002A        PRINTER CONTROL WORD(PORT C)
                               *PRDAT EQU    200000A        PRINTER DATA(PORT A)
                               *
    137.165                          ORG     137165A        UPPER 140 BYTES FOR HANDLER
                               *
                               *
                               * HANDLE PRINTER ECHO
                               *
    137.165   076 251          LPDRV  MVI    A,251Q         PIA SET UP PORT A MODE 1 OUTPU
T
                               *                                     PORT C MODE 0 INPUT

V   137.167   062 003 200            STA     200003A
V   137.172   072 002 200            LDA     200002A         GET PRINTER CONTROL
    137.175   346 040                ANI     40Q            IF SW1 LOW,NO PRINT
    137.177   312 233 137            JZ      ENDPRT
V   137.202   072 002 200      L1     LDA    200002A
    137.205   346 204                ANI     204Q           CHECK OBF/ACK
    137.207   356 204                XRI     204Q
    137.211   302 202 137            JNZ     L1             LOOP UNTIL BOTH ARE HIGH
    137.214   361                    POP     PSW            GET CHAR TO PRINT
V   137.215   062 000 200            STA     200000A         PRINT IT
    137.220   365                    PUSH    PSW            SAVE CHAR FOR CRT
V   137.221   072 002 200      L2     LDA    200002A
    137.224   346 204                ANI     204Q           CHECK OBF/ACK
    137.226   356 204                XRI     204Q
    137.230   302 221 137            JNZ     L2             LOOP UNTIL BOTH HIGH
                               *
                               * NOW DO CRT OUTPUT
                               *
V   137.233   072 002 200      ENDPRT LDA    200002A
    137.236   346 020                ANI     20Q            IF SW2 LOW NO CRT OUTPUT
    137.240   312 247 137            JZ      ENDCRT
    137.243   361                    POP     PSW            GET CHAR
    137.244   303 111 040            JMP     $CDOUT         GO OUT TO CRT AND RETURN
    137.247   361              ENDCRT POP    PSW            CLEAR STACK
    137.250   311                    RET                    RETURN IF NO CRT OUT
                               *
                               * FROM HERE TO PHYSICAL END OF MEMORY MUST LEAVE SPACE
                               *      FOR PAM TO SET UP ITS STACK POINTER ON RST/0
                               *      NOTE THAT PAM DOES NOT RESPECT HIGH MEMORY AS
                               *      CONFIGURED IN OTHER HEATH SOFTWARE........
                               *
    137.251                          END       START
```

Driver source listing.

Power AND Flexibility for the HEATH® H8



D-G Electronic Developments Co. introduces NEW Hardware, Firmware, and Software support for the H8. Combined with the already popular DG-80 CPU, our products provide an ever increasing line of complimentary support devices to enhance the power and flexibility of the H8 Computer.

INTRODUCING: the **DG-64D**

- Up to 64K bytes capacity Dynamic RAM
- Hardware bank selectable in 8K increments
- Software bank selectable in 16K increments through I/O port
- On-board bank select/CPU ROM disable port, addressable to any 256 I/O addresses
- Up to 8 boards controllable through one I/O port (allows page mode operation)
- On-board transparent refresh for 8080 or Z80 microprocessor backed up by asynchronous refresh upon loss of normal program execution
- 4 MHz operation with no wait states required
- Low power consumption
- Assembled, tested, & burned-in—90 day warranty

Prices:

64K 529.00

48K **480.00**

32K **431.00**

16K **382.00**

ØK **333.00**

Documentation only (DG-64D) **15.00**

STATE OF THE ART CPU FOR THE HEATH® H8

DG-80 Z80® CPU—249.00 (Assembled & Tested) Documentation only: $25.00

NEW—SUPPORT for the DG-80

the DG-FP8—69.95

Monitor/Utility package for use with the DG-80 CPU which provides functions of PAM-8 as well as the following:

- Split octal or hexadecimal display and entry
- Two keystroke display of memory contents pointed to by any register
- Sets PC register to boot address on power-up
- Display and alteration of Z80 primary and alternate registers
- Provision for Z80 non-maskable interrupt
- Provides front panel single step
- Real time clock
- Supports standard CP/M® provided by DG Electronic Developments Co. as well as HDOS

DG-FP8/DG-ADP4—TOGETHER—$79.95

Documentation Only — $15.00 (Source Code Not Included)

the DG-ADP4—19.95

Plug-in hardware modification to allow operation of the Heath H17 disk system with the DG-80 at 4 MHz. Requires the use of the DG-FP8 firmware package.

DG-CMD1—$29.95

ROM disable port for use with the Heath® H8 computer. Addressable to any of 256 I/O ports. Allows the use of a full 64K of RAM when used in conjunction with the DG-80 CPU and the DG-FP8 hardware/firmware package (NOT REQUIRED FOR SYSTEMS UTILIZING THE DG-64D MEMORY BOARD).

STANDARD CP/M® Ver. 2.2—$130.00

16K CHIP SETS (8-4116 Type Dynamic RAMS) for DG-32D, Apple®, TRS-80®, H88/89® and PET® (Tested) ... **$49.00**

NEW PRICING ON OUR POPULAR DG-32D (32K Dynamic RAM for Heath H8)

32K 339.00

16K (½ populated) 287.00

ØK 235.00

Documentation Only 12.00

D·G ELECTRONIC DEVELOPMENTS CO.

145

Ordering Information: Products listed available from DG Electronic Developments Co., P.O. Box 1124, 1827 South Armstrong, Denison, Tx. 75020. Check, Money Order, VISA or Master Charge accepted. Phone orders (charge only) call (214) 465-7805. No COD's. Freight prepaid. Allow 3 weeks for personal checks to clear. Texas residents add 5%. Foreign orders add 30%.

A Mini Logic Monitor And Single-Cycler For Hardware Debugging

This addition to your test bench helps isolate elusive glitches.

Wayne D. Smith, Ph.D.
Austin Peay State University
Clarksville, TN 37040

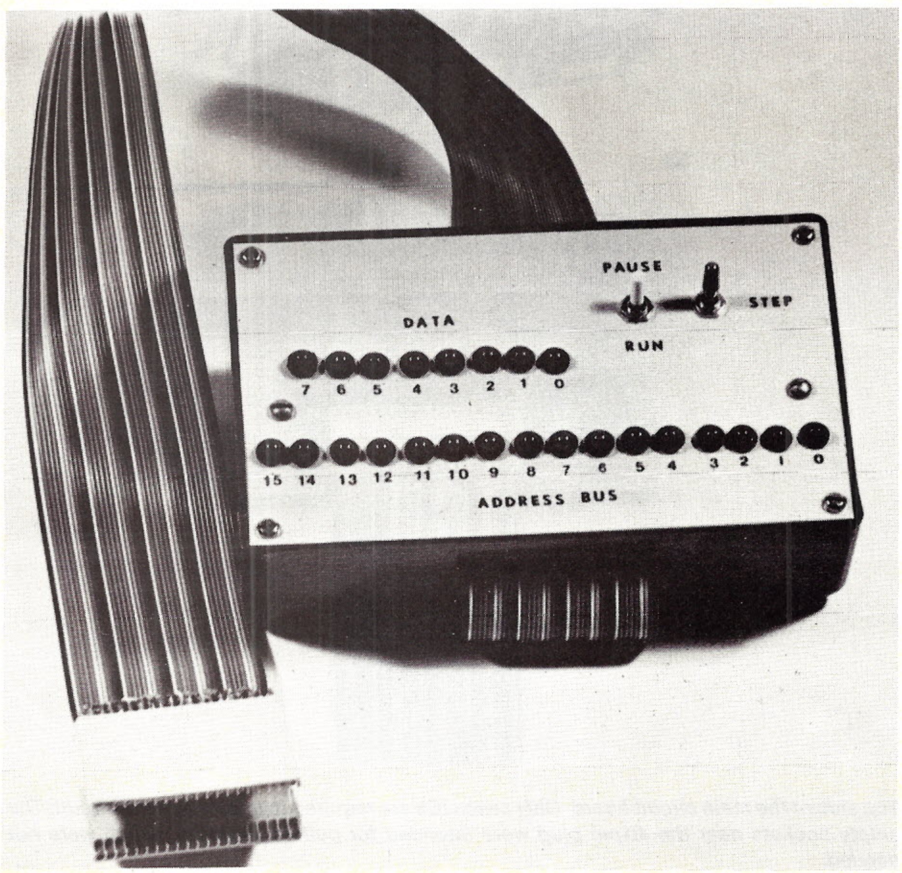
If you are lucky in scratch-building a dedicated microcomputer system, the system will function properly on the first reset. Unfortunately, it is far more common for the system to fail to function. This failure can be partial, that is, the system functions in some unintended manner, or, more frequently, the system may give absolutely no external indications of any system function at all.

The complexity of microcomputer systems makes such occurrences both commonplace and exasperating. Trying to locate the problem when there are no external indications of circuit operation can take many hours. You must investigate various problem areas: the clock circuit, the reset circuit, miswired address or data lines, incorrect address decoding, improper or noisy power supply or incorrect software. If you have a large system, just determining the general area to investigate can take quite some time.

I teach an advanced microcomputer architecture course at the NASA Johnson Spacecraft Center, where the students design, wire and test from one to three small special-purpose microcomputer systems. With from eight to 12 systems under construction at any one time, debugging such systems is impossible without some method for limiting the areas to be checked for errors. I devised a test circuit that can be added to the system under test to help isolate problem areas.

Design Considerations

In designing the test circuit, I had to con-

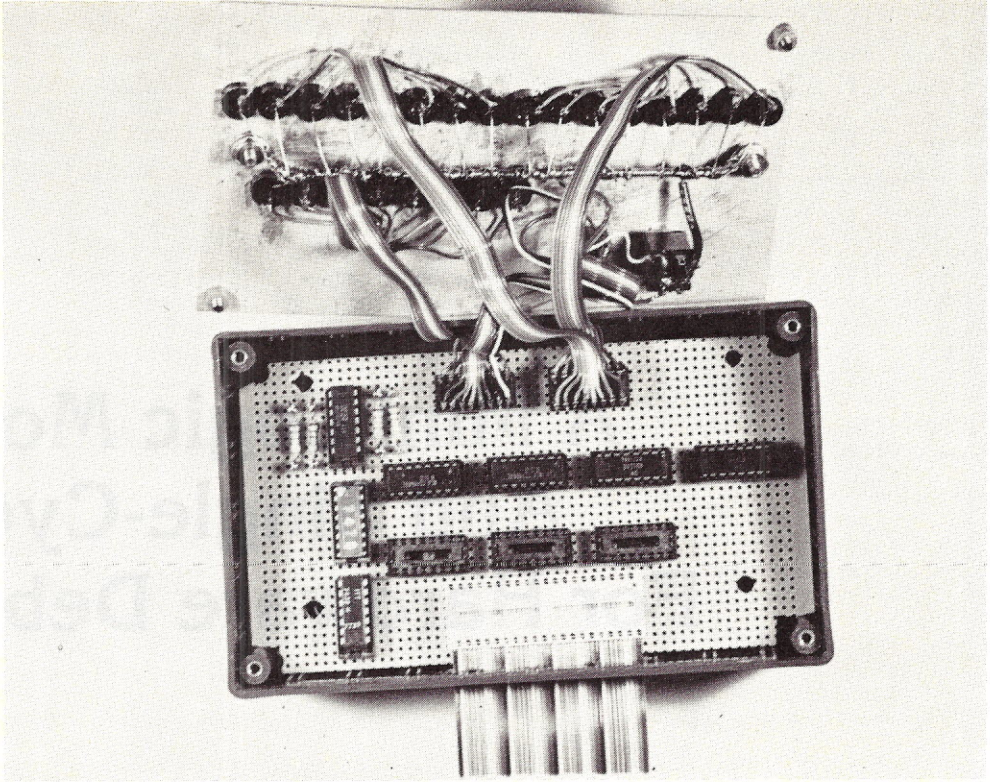


Exterior view of the single-cycle circuit, which is completely self-contained and even draws its power from the microprocessor under test. A single 40-pin proto-clip connects the circuit to the microprocessor.

sider several constraints: The test circuit should not interfere with the normal operation of the circuit under test, and the student should not have to perform extensive rewiring of his system in order to insert or remove the test circuit, which should be portable to facilitate movement from one system to another. The circuit should also be simple to operate, since many of the students would be unfamiliar with complex test equipment such as logic analyzers. And, of course, the circuit should be inexpensive enough to permit providing several such units in the laboratory.

A circuit to track down system problems should include some method for reducing system speed so you could observe operation. At the same time, provisions would be made for supplying you with pertinent information about what the system is doing at any particular time. The obvious solution would be some method for slowing the system clock to a frequency that would permit direct observation of system operation.

Unfortunately, this approach is not practical. Most of the internal registers of a microprocessor are dynamic in nature and require a minimum clock frequency in order to ensure proper refreshing. The minimum frequency of the 6502 processor used in this course was experimentally determined to be about 200 kHz. Below this frequency, circuit operation became erratic and unreliable. Even this relatively slow speed is still much too fast to allow you to observe the circuit.



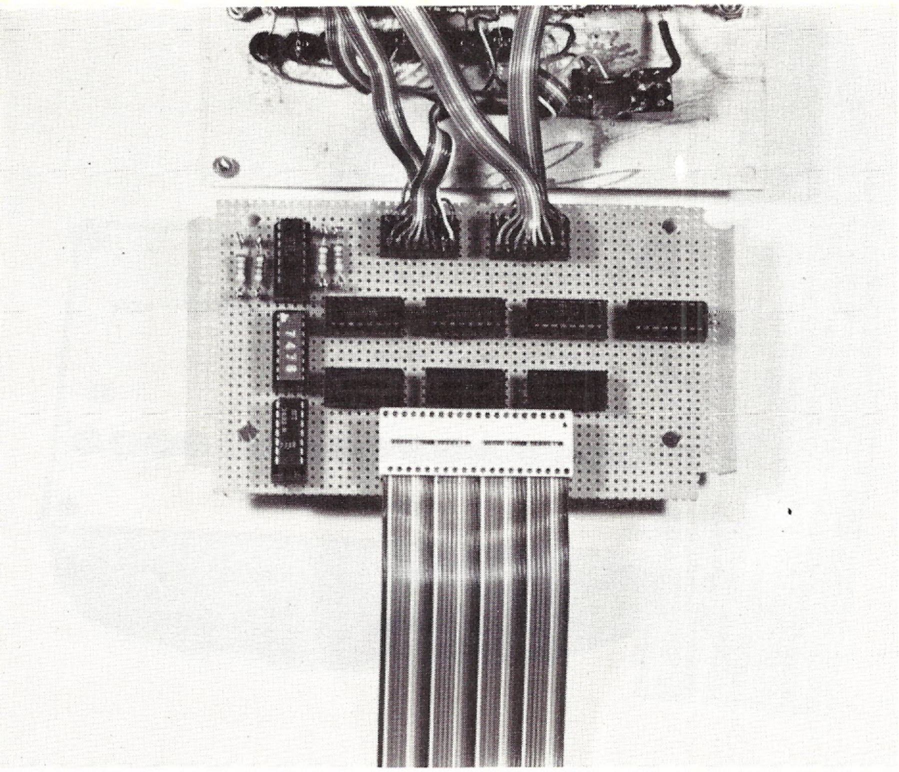
Interior detail of the main circuit board for the single-cycle circuit. The board is connected to the front panel and the microcomputer through ribbon cable. Both cables connect to sockets on the main circuit board for ease of assembly.

Another method for reducing the speed of a microprocessor involves the use of the ready input to the processor. This input is normally used when the processor system is equipped with memory that has a cycle time longer than that of the processor. When the ready input goes low, the processor is placed into a wait state, where it will cycle continuously in the memory read phase until the ready signal is high at the beginning of the cycle.

When the ready line goes high, the processor will accept the input on the data bus and continue with normal operations. During the time the processor is waiting for the ready line to go high, the address bus is held constant by the processor to allow external decoding of the address. At the same time, the data bus is maintained in the input or read mode. Any value that appears on the data bus during this period originates from external devices.

The ready input can also be used to perform single-cycle operation in a microcomputer. To accomplish this type of operation, it is only necessary to ensure that the ready input remains low most of the time. This causes the processor to remain in the wait state until you advance to the next cycle by providing a short, high pulse on the ready input and then bring the input low again. The exact duration and timing of the high pulse on the ready line can be obtained by synchronizing the momentary high signal with the Ø2 clock signal.

Since the processor will maintain the address bus constant while the ready signal is low, you can examine the bus to determine the effective address of the operation tak-



Top side of the main circuit board. Only seven ICs are required to implement the circuit. The empty sockets near the 40-pin plug were intended for pull-up resistors, which were not needed.

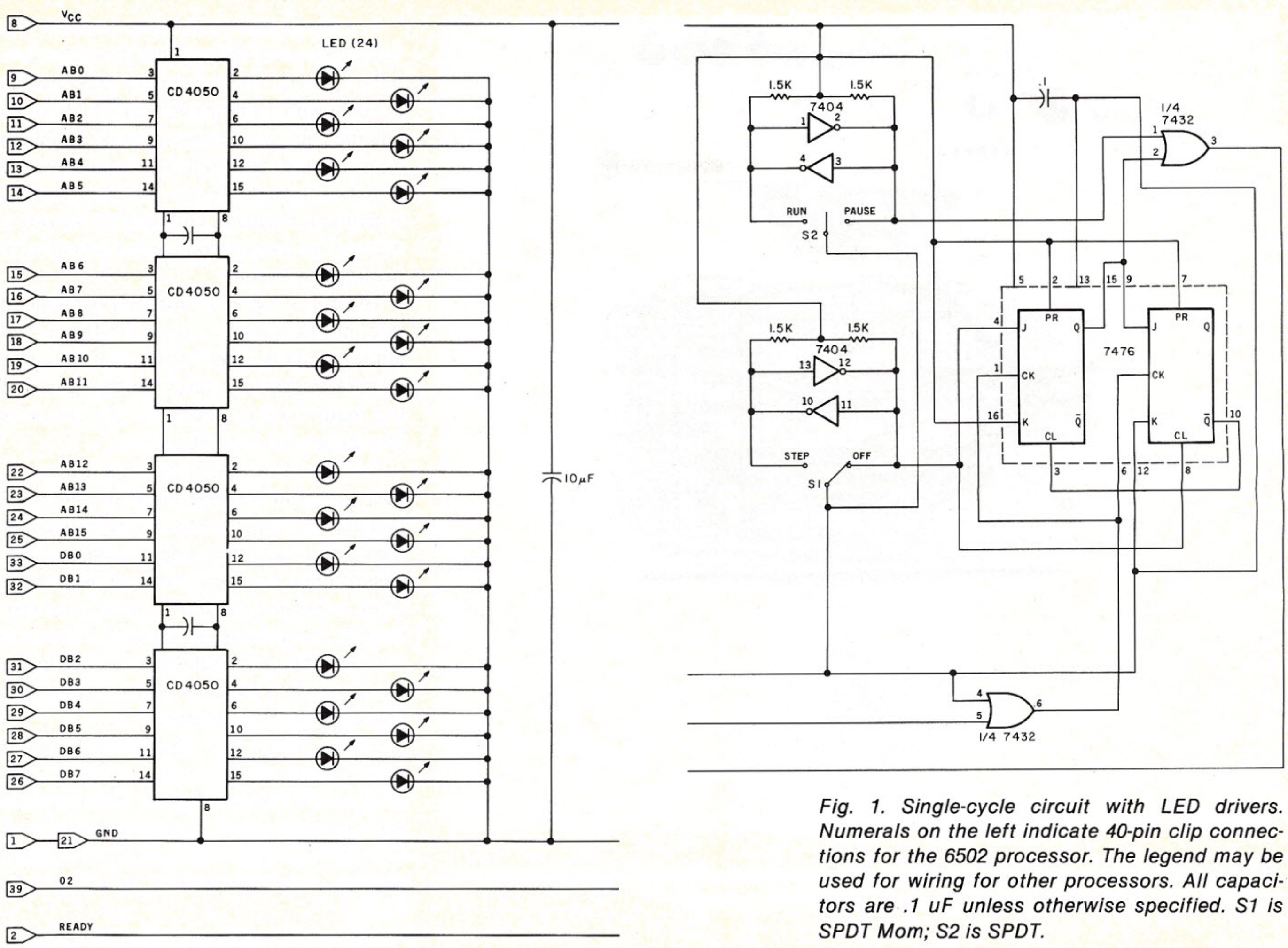


Fig. 1. Single-cycle circuit with LED drivers. Numerals on the left indicate 40-pin clip connections for the 6502 processor. The legend may be used for wiring for other processors. All capacitors are .1 uF unless otherwise specified. S1 is SPDT Mom; S2 is SPDT.

ing place with either a voltmeter or a logic probe. Similarly, you can examine the data bus to determine the values being placed on this bus by the external devices. Address decoding signals can also be checked while in the wait state.

Two Points

This process is a single-cycle, rather than single-step, operation. In single-cycle operation, each separate cycle of the instruction is stopped. In a typical machine instruction, the machine would be stopped for the operation code fetch, then stopped for one or two address fetches and finally stopped during instruction execution, provided the instruction is a memory read.

During a single-step-type operation, the processor performs the complete fetch and execution of a single instruction before stopping. The machine usually stops during an operation code fetch, which is the only operation that can be examined in detail.

Another salient point is that using the ready input only allows stopping the processor when memory is being read. This includes instruction op code and address fetch, memory-to-register instruction execution, input operations and stack pops.

Since the ready signal is not tested by the processor during write operations, these operations proceed (after the fetch of the instruction and its associated addresses) at normal processor speed. Internal register operations, such as clearing the carry flag, also proceed at normal speed.

While single-step operation is valuable in debugging software, it is less useful for finding errors in system hardware, especially in new, untried systems. The circuit described in this article provides for single-cycle operations, and has proven to be quite valuable in debugging systems hardware. My experience with this circuit indicates that the errors can usually be found, even without the ability to stop memory-write operations.

You can make provisions to capture the signals involved in memory-write operations, even though you can't stop these operations.

A Solution

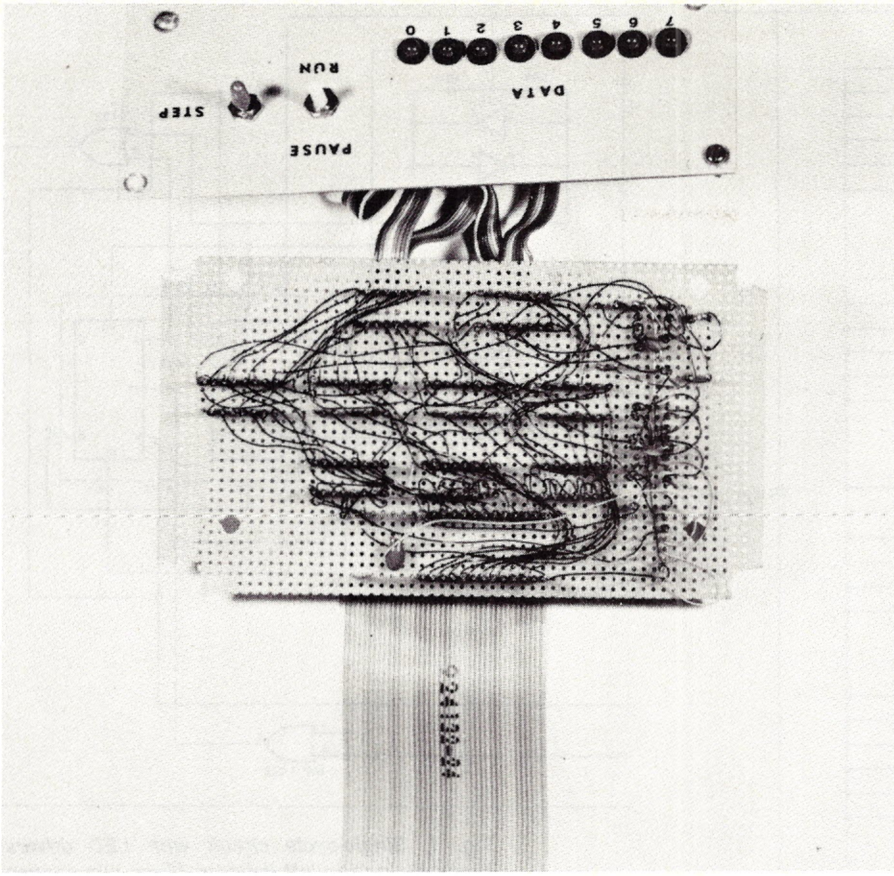
The single-cycle circuit shown in Fig. 1 is designed for use in debugging hardware systems. The heart of the circuit is the two J-K flip-flop pulse synchronizing circuit, which synchronizes the output of a push-button switch with the Ø2 clock signal. The output of the synchronizer remains low most of the time.

When the pulse switch is moved to the step position, a single, high pulse is generated and is synchronized with the next full occurrence of a high Ø2 clock signal. The output then returns low and will remain low until the switch is released and then depressed again.

The output of the synchronizer circuit is ORed with a run/pause switch to allow either single-cycle or full-speed operation. When this switch is in the run position, the output to the ready input is high, allowing full-speed operation of the processor. With the switch in the pause position, the output of the OR gate is determined by the output of the synchronizer circuit.

The synchronizer circuit requires input from a debounced switch for proper operation. The two inverters associated with the step switch provide this switch debouncing. A similar circuit is used to debounce the run/pause switch to ensure a smooth transition from one mode of operation to the other.

In order to make the circuit self-sufficient, the address and data bus lines are connected directly to LED drivers. This adds slightly to the cost of the circuit, but eliminates the need for a logic probe and



Bottom view of the main circuit board showing wire-wrap details. Note three despiking capacitors. The three sockets near the 40-pin socket are not used.

speeds up system testing considerably. CMOS CD4050 buffers are used for driving the LEDs from the data and address lines.

The only external input to the basic circuit is the microprocessor Ø2 clock signal. In my implementation, both the run/pause and the step switches are built into the debugging system. The 16 address lines and the eight data lines are connected directy to the CD4050 chips, which are used as LED drivers. The debugging circuit imposes one TTL load on the Ø2 line and one CMOS load on each of the address and data lines. In practice, two loads on the Ø2 line have been acceptable, and the OR gate buffer for this signal may be omitted if desired.

In a permanent system, the circuit in Fig. 1 could be wired directly into the microcomputer system. This would not only provide the user with a single-cycle capability when desired, but would also provide "blinking lights" to impress visitors when the system is in the run mode. To use in a laboratory environment, however, you have to provide a method for quickly connecting and disconnecting the circuit.

The quick-connect capability is provided through the use of a 40-pin proto-clip. The entire circuit is wired to the clip, including all data and address lines, the Ø2 signal and the ready output. Power for the circuit is also drawn from the 40-pin clip.

By assembling the circuit in this manner, you can install the test circuit by simply attaching the clip directly to the processor itself. You can then single-cycle the processor until you locate the problem area. After correcting the error, you can remove the clip, and processor operation returns to normal. Placing the run/pause switch in the run position will also enable full-speed operation of the processor.

Before using the clip, remember that when the ready signal is not used in a system, it is normally tied to Vcc. In order to allow the test circuit to pull the ready input low, it is necessary to make this connection through a pull-up resistor of about 1.8k ohms.

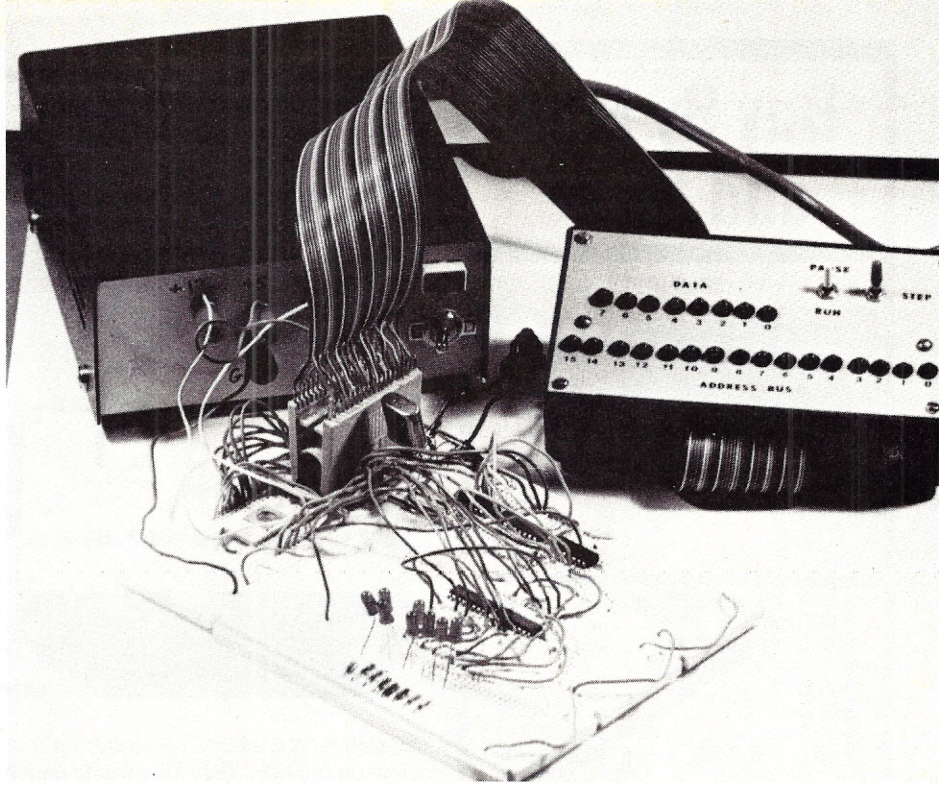
If the circuit to be tested has the ready input tied directly to Vcc, it will be necessary to break this connection and replace it with the resistor. Installing this resistor has no effect on the operation of the processor when the test clip is not attached.

If the system under test already has other connections to the ready line—as would be the case when slow memory is used—addi-

tional wiring will be required. It will be necessary to OR the current ready inputs with the output from the test circuit. This will require slightly more wiring, but will provide the single-cycle operation when needed. When the test clip is not attached, the test circuit OR gate input should be connected to Vcc through a 1.8k ohm resistor.

Construction of the circuit is not critical, although several despiking capacitors (0.1 uF) should be provided. Additionally, a 10 uF tantalum capacitor should be provided where the power supply leads enter the circuit. These are precautions to compensate for the long cable length between the processor and the circuit. These capacitors are indicated in Fig. 1. Resist the temptation to use 74LS circuits, since I have had some difficulty with noise when using the 'LS types.

I designed the circuit specifically for use with laboratory systems using the 6502 processor. However, you can easily adapt it to any microprocessor that has a ready input. It has been used successfully with 8080 systems, although it may be necessary to use more than one clip to accommodate clock-driver and bus-controller chips on some systems. It has also been used to diagnose a malfunctioning KIM-1. Use with the KIM-1 system requires no modification of the KIM, since the ready pull-up resistor is supplied on the KIM board.



The single-cycle circuit connected to a typical student microcomputer design. The 40-pin proto-clip is the only interconnection between the two circuits. Removing the clip will restore full-speed operation to the microcomputer system.

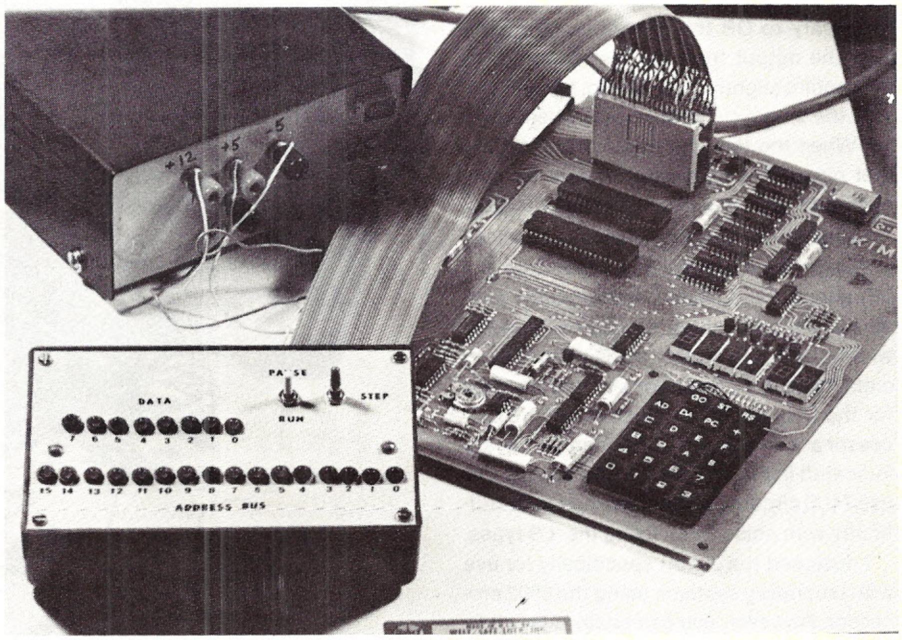
Using the Circuit Output

The basic function of the test circuit is to allow you to single-cycle a malfunctioning microcomputer system in order to isolate hardware problems. As the circuit is cycled through various operations, you can observe the values placed on the address bus by the processor at each step. You can also observe the values that are placed on the data bus by memory and input units. Usually, this information is sufficient to pinpoint problem areas.

Interpreting the values placed on the buses requires a little practice, however. One feature that complicates matters is the microprocessor practice of pipelining instruction fetches. Most eight-bit processors work with variable-length machine-language instructions. That is, an instruction may consist of one, two or three bytes, depending on the operation code.

For example, the CLC instruction in the 6502 is a one-byte instruction. The BEQ instruction is two bytes, and the JMP instruction is three bytes. This means that during the instruction fetch cycle, the processor may be required to access memory either one, two or three times.

In order to optimize the fetching of variable-length instructions, most processors utilize the technique of pipelining, which is predicated on the idea that a two-byte instruction may be thought of as an



The single-cycle circuit connected to a KIM-1 microcomputer. No modifications are required on the KIM, and I have found that this arrangement makes an excellent classroom demonstration of microcomputer functions. It has also been used to diagnose malfunctioning KIMs.

average instruction. Therefore, the processor is set up to always execute a two-byte fetch.

After the first byte is fetched, the processor proceeds with fetching a second byte, while the first byte (operation code) is being decoded in the CPU. By the time the operation code is decoded and the correct number of bytes determined, the second byte is already stored in the address portion of the instruction register in the CPU.

If the operation code is, in fact, a two-byte instruction, the processor proceeds with the execution of this instruction, since both bytes are already available in the CPU. If, on the other hand, the instruction calls for three bytes, then execution is deferred while the processor fetches the third byte. Execution of the instruction then proceeds for the three-byte instruction.

Sometimes, however, the operation code fetched during the first cycle indicates a one-byte instruction. In this case, the processor has already fetched a second byte, but doesn't need it. When this happens, the data fetched during the second cycle is ignored by the processor. However, the program counter is decreased by one to ensure that the second byte fetched during this operation becomes the first byte fetched during the next cycle.

Although this process may seem complex to the beginner, it is actually quite efficient. The operation code must be decoded in any event, and during this time, the data and address buses are not needed by the processor for other purposes. Utilizing this decoding time to fetch the second byte represents efficient use of processor facilities.

If a two-byte instruction is called for, then the processor is ready to proceed as soon as the decoding is complete. If a three-byte instruction is called for, then the processor is already two-thirds through the fetch cycle. Even in the case of a one-byte instruction, no time is lost, because the program counter can be decremented while the one-byte instruction is being executed. Since two-byte instructions predominate most programs, the processor is geared to process this type of instruction at maximum speed.

Pipelining

Pipelining leads to one of the most confusing aspects of programming microprocessors in machine language: the reversal of the two bytes of an address field in a three-byte instruction. This requirement is brought about by the nature of the normal two-byte fetch used in pipelining.

When an instruction is only two bytes long, the second byte is usually placed on the low-order eight bits of the address bus during instruction execution. Therefore, when the second byte is fetched by the processor, it is placed in the low-order part of the instruction register address field. Since it would take a complete machine cycle to move this byte to the high-order address field, it is much faster to simply leave it where it is when a third-byte fetch is required. Therefore, programmers reverse the address bytes of a three-byte instruction in order to reduce the instruction execution time by one machine cycle.

All of this leads back to the interpretation

of the values displayed on the buses during the use of the single-cycle circuit. Without a basic knowledge of pipelining, you would not understand many of the values displayed. It is a common occurrence to note the fetch of a second byte, even though you know that the operation code fetched during the first cycle was a one-byte instruction.

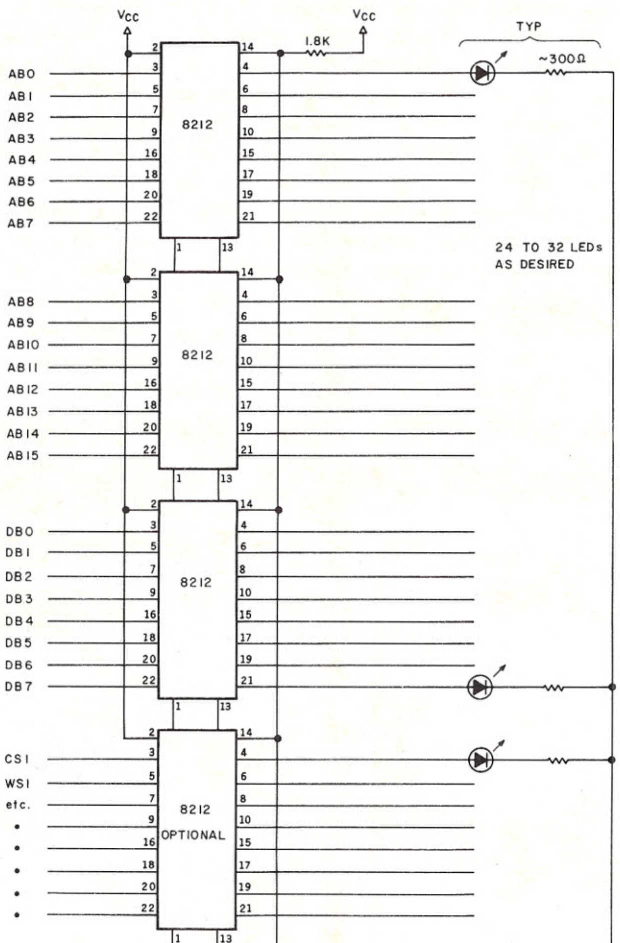
In order to fully utilize the circuit under discussion, you must appreciate the concept of pipelining and have access to information about the cycles utilized by your machine in fetching and executing various instructions. For the 6502 processor, this information is well presented in appendix A of the hardware manual. In some other systems, it may be necessary to extrapolate this information from timing diagrams furnished by the manufacturer. In any event, a little practice with an operational system will prepare you to use the single-cycle circuit to diagnose malfunctioning units.

Be warned that the foregoing discussion of pipelining is an oversimplification. In a one-byte instruction, the processor actually decodes and executes the instruction while the second byte is being fetched. The second byte is then fetched again as an operation code for the next instruction. Special addressing modes, such as immediate, indirect and indexed, greatly complicate the process. However, the simplified explanation serves to make the point: To obtain full benefit from single-cycle operations, the appropriate references for a specific processor must be consulted.

Solving the Output Problem

The circuit works well as described; in all cases encountered to date, the circuit has been sufficient to isolate hardware problems for correction. There is, however, always the possibility that a problem that only arises during the execution of an output or memory-write instruction may exist. The circuit will not allow the detection of errors in the execution of this type instruction, since the processor cannot be halted during execution of these instructions through the use of the ready input.

It is impossible to stop the processor during the execution of a memory-write instruction. There is, however, a way to record



MPU	$\overline{DS1}$	DS2
6502	R/W	$Ø_2$
8080	WR	Vcc

Fig. 2. Snapshot circuit to record data values involved in processor write or output operations. The first three 8212s capture values from the data and address buses. The fourth chip captures other values of interest.

While the single-cycle circuit does not make child's play out of hardware systems debugging, it does greatly simplify the task of finding errors in a microcomputer system. My daughter, Erin, doesn't exactly understand what it all means, but she enjoys watching the flashing lights on the circuit.

what the processor does during one of these operations. The circuit to "snapshot" the data and address buses during any write operation is shown in Fig. 2.

Essentially, the circuit detects any memory-write operation and latches the contents of the data and address buses into D-type flip-flops at the instant that the write takes place. Since the single-cycle circuit will halt the processor before the next instruction fetch operation takes place, the user can examine the values that were on the buses when the write took place.

Considering the expense of this circuit, with three 8212s and 24 more LEDs, it might not seem worth the effort. Furthermore, the circuit does not retain any of the other signals that are associated with a write operation. It may be necessary to use a fourth latch to "snapshot" chip-enable and write-select signals that are used during the write operation. A more cost-effective approach is to install just the latches, with several unassigned inputs for additional signals, and then use a logic probe to determine the latch contents after the write takes place.

If a scratch-built processor fails to function properly and if the single-cycle circuit fails to locate the error, you may have to resort to this "snapshot" circuit. Fortunately, I have not yet had to use this circuit in the laboratory. However, I keep a board of 8212s wired up, just in case.

Summary

The problem of debugging a microcomputer hardware system presents a significant challenge. This circuit will give you the capability of easily installing a test circuit that will assist in locating the area of the processor system where a problem may exist. Normal troubleshooting methods can then be used to isolate and correct the error. The use of the single-cycle circuit can significantly decrease development time, especially in a laboratory environment where students may have little experience with more advanced diagnostic equipment.

There are several other uses for a single-cycle circuit: It can be wired directly into a processor system to provide an inexpensive "front panel" that can permanently remain in the system. This panel is useful in locating software errors. For example, a tight program loop will give some LEDs the appearance of being permanently on. A longer loop will look like several distinct patterns on the LEDs. Cassette tape loads can also be monitored by observing the pattern on the LEDs.

The circuit also has applications in teaching computer architecture and microcomputer hardware courses. Students can gain a solid understanding of machine cycles by single-cycling the machine through a short program. Resets, subroutines and interrupts are also easily demonstrated with the circuit. The circuit can also be used to diagnose turnkey systems. The interested reader can probably discover several other applications. ■

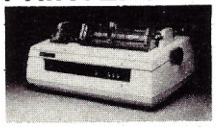
WE WILL NOT BE UNDERSOLD

16K MEMORY UPGRADE KITS 2 for $85 $45

for TRS-80*, Apple II, (specify): **Jumpers** $2.50

PRINTERS

NEC Spinwriter



Letter Quality High Speed Printer
Includes TRS-80* interface software, quick change print fonts, 55 cps, bidirectional, high resolution plotting, graphing, proportional spacing: R.O. **$2550**

R.O. with Tractor Feed **$2650** KSR with Tractor Feed **$2950**

779 CENTRONICS TRACTOR FEED PRINTER **$969**
Same as Radio Shack line printer I

737 CENTRONICS FRICTION & PIN FEED PRINTER **$799**
9 x 7 matrix Same as Radio Shack line printer IV

730 CENTRONICS FRICTION & PIN FEED PRINTER **$629**
7 x 7 matrix Same as Radio Shack line printer II

P1 CENTRONICS PRINTER Same as Radio Shack quick printer **$269**

PAPER TIGER (IP440) Includes 2K buffer and graphics option **$879**
(IP460) Bidirectional, 160 cps, graphics and 2K buffer **$1075**

TI-810 Faster than Radio Shack line printer III. Parallel and serial w/TRS-80* interface software w/u + l case & paper tray **$1589**
Compressed print, vertical form control **$1865**

OKIDATA Microline 80 Friction and pin feed **$549**
Tractor Feed, friction, and pin feed **$649**

Microline 83 Bidirectional, 120 cps, uses up to 15″ paper **$1050**

EATON LRC 7000 + 64 columns, plain paper **$299**

ANADEX **DP-9500** **$1359** **DP-8000** **$825**

CAT MODEM Works same as Radio Shack Telephone Interface II **$148**

LEEDEX MONITOR Video 100 **$119**

ZENITH Color Monitor **$379**

SANYO Model VM 4509 9″ Monitor **$155**

DISK OPERATING SYSTEMS

PATCHPAK #4 by Percom Data **$ 8.95**

CP/M® for Model I, Zenith **$145** • for Model II, Altos **$169.00**

NEWDOS Plus **40track** **$ 99.00**

NEWDOS 80 **$135.00**

ACCESSORIES

HEAD CLEANING DISKETTE: Cleans drive Read/Write head in 30 seconds. Diskettes absorb loose oxide particles, fingerprints, and other foreign particles that might hinder the performance of the drive head. Lasts at least 3 months with daily use. Specify 5¼″ or 8″. **$20 ea/$45 for 3**

FLOPPY SAVER: Protection for center holes of 5¼″ floppy disks. Only 1 needed per diskette. Kit contains centering post, pressure tool, tough 7-mil mylar reinforcing rings. Installation tools and rings for 25 diskettes. **$ 11.95** **Re-orders of rings only** **$ 7.95**

EXTERNAL DATA SEPARATOR: Eliminates data separation problems (crc). Improves reliability. This plug in unit comes fully assembled and tested. **$ 29.95**

RS232 **$ 84.00**

TRS232: Teletype current loop output from cassette port **$ 49.00**

DISK-DRIVE EXTENDER CABLES: Fits all mini-disk drives. **$ 16.95**

SIX (6) PRONG ISOLATOR: ISO-2 **$ 54.00**

AC FILTER/6 PRONG POWER STRIP **$ 39.00**

DISK DRIVE CABLES: 2 drive **$29.00** 4 drive **$ 35.00**

DUST COVERS: TRS-80/Apple **$ 7.95**

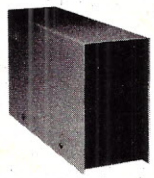
PLASTIC DISKETTE HOLDER: For ring binder, holds 20 **$ 8.00**

RF MODULATOR: Adapts video to TV **$ 35.00**

TRS-80 & OTHER MYSTERIES **$ 18.95**

NEC SPINWRITER THIMBLE **$19.95** **RIBBON** **$ 6.95**

CCS CARDS: Parallel or serial **$115.00**



DISK DRIVES $314

40 track, 102K Bytes. Fully assembled and tested. Ready to plug-in and run the moment you receive it. Can be intermixed with each other and Radio Shack drive on same cable. TRS-80* compatible silver enclosure. 90 day warranty. One year on power supply. **External card edge included.**

FOR TRS-80*		
CCI-100	5¼″, 40 Track (102K Bytes) for Model I	**$314**
CCI-280	5¼″, 80 Track (204K Bytes) for Model I	**$549**
CCI-800	8″ Drive for Model II (½ Meg Bytes)	**$795**
For Zenith Z89		
CCI-189	5¼″, 40 Track (102K Bytes) add-on drive	**$394**
Z-87	Dual 5¼″ add-on drive system	**$995**

DISKETTES — Box of 10 (5¼″) - with plastic library case **$24**
8″ double density for Model II (box of 10) **$36**

COMPLETE SYSTEMS

ALTOS 64K, DD, SS, 2-Drive, 1MB	**$3995**
APPLE 16K	**$989**
TRS-80* Model II-64K	**$3499**
TRS-80* LEVEL II-16K with keypad	**$689**
TRS-80* Expansion Interface	**$249**
HEWLETT PACKARD HP-85	**$2950**
ZENITH Z89, 48K all-in-one computer	**$2500**
ZENITH Z19	**$735**

TELEVIDEO **912B $745** **912C $755** **920B $769** **920C $779**

ATARI 400 $489 **ATARI 800 $749**

APF **Game Only $99** **Complete System $499**

MATTEL INTELLIVISION **$229**

Software available for all Complete Systems

SOFTWARE FOR THE TRS-80* Software w/Manual

INTELLIGENT TERMINAL SYSTEM ST-80-III BY LANCE MIKLUS: Enables a TRS-80* to act as a dial-up terminal on any standard time sharing network. Provides a TRS-80* with control key, ESC Key, Repeat Key, Rub Out Key, Break Key, full upper and lower case support, selectable printer output and program selectable transmission rates **$139**

CCA-DATA MANAGEMENT SYSTEM: Automate your information processing tasks. You can create a file of customer information; quickly and easily add, delete or update records; search a file; keep a file in order of the value in any field; and print records and labels in any desired sequence or from just a part of a file. Requires 32K TRS-80 and one drive. **$72.00**

S & M SYSTEMS
INSEQ-80™: Indexed Sequential Access Method (ISAM) for the TRS-80 Model I. A must for anyone writing business programs. Eliminate wasted disk space from direct record processing. Split second access to any record. Access data records instantly via alpha/numeric "key" eg. Part NR, zip code or sequentially in ascending key sequence. Add/modify records in any order. Access up to three files per program — Files may be spread over multiple disks. Machine language processing from your basic program. Utility program to convert direct files to INSEQ-80 format. **$49.00**

FULLY INTERACTIVE ACCOUNTING PACKAGE: ISAM (INSEQ-80) based. Includes General Ledger, Accounts Payable, Accounts Receivable and Payroll. System runs "stand alone" or "co-ordinated G/L" at users option. Based on Osborne accounting method. Requires 32K, TRS-80, 2 or 3 drives. N/A CA.

General Ledger	**$99**
Accounts Receivable	**$99**
Accounts Payable	**$99**
Payroll	**$99**

Osborne books: Req'd as additional documentation $20 ea

INVENTORY Requires 32K, TRS-80, 1 drive **$125**

INSORT-80: Callable form BASIC via USR. Sorts "Random" Disk Files. "Disk" to "Disk" sort times — 350 records in 35 secs, 1000 records in 6 minutes, 3500 records in 12 minutes. Machine language processing. Up to 35 sort keys ascending/descending. Utility to build BASIC program. Runs under NEWDOS. **$49.95**

CP/M® BASED SOFTWARE for Zenith, Altos, Radio Shack, Apple Software w/Manual

Z-80 SOFTCARD FOR APPLE: Your key to future software expansion. Get the best of both worlds, Apple's 6502 and CP/M Z-80. Plug in the card and get a Z80. Supports Apple language card and all Apple peripherals. Comes with set of three manuals. **$339**

CCI-TELNET VERSION 5: A communication Package which enables microcomputer users to communicate both with Large Mainframes and other microcomputers. Extensive commands make it useful in many applications where communication between computers is necessary. Powerful terminal mode enabling user to save all data from a session on disk. Completely CP/M compatible. Multiple communication protocols supported. Able to transfer files in both directions without protocol where the other machine does not support any protocol. Extensive ON-SCREEN help. Source code provided. **$149**

MICROPRO-WORD-STAR: Menu driven visual word processing system for use with standard terminals. Text formatting performed on screen. Facilities for text paginate, page number, justify, center and underscore. User can print one document while simultaneously editing a second. Edit facilities include global search and replace. Read/Write to other text files, block move, etc. Requires CRT terminal with addressable cursor positioning. **$399**

DEALER (NATIONAL/INTERNATIONAL) INQUIRIES INVITED **Send for FREE Catalogue**

The CPU SHOP ✓256

TO ORDER CALL TOLL FREE 1-800-343-6522
Massachusetts residents call **(617) 242-3361**
For detailed technical information, call 617/242-3361
Hours: 10AM-6PM (EST) M-F (Sat. till 5)

5 Dexter Row, Dept. K11M
Charlestown, Massachusetts 02129

Massachusetts residents add 5% sales tax

Quantities on some items are limited

VISA®

master charge

*TRS-80 is a Tandy Corporation Trademark ® Digital Research

Efficient Data Storage For Microsoft BASIC

Send your single- and double-precision numbers packing.

James Monagan
806 Clark St.
Iowa City, IA 52240

Microsoft BASIC includes several functions for packing and unpacking numeric data, primarily to permit efficient data storage in random disk files. These functions allow integer, single-precision and double-precision numbers to be packed into strings that are two, four or eight bytes long. Without packing, the numbers would have to be stored as ASCII character strings that could be up to 18 bytes long.

Where random file records are fixed in length, as in Altair BASIC or TRS-80 BASIC, it often takes much effort to partition the buffer so that all data items will fit in a single record. Even where records may have an arbitrary length, as in BASIC-80, it is usually important to keep them as short as possible. Shorter records mean more records can be stored per diskette.

The techniques described in this article let you pack most single- and double-precision numbers of interest into three- or four-byte strings instead of four- or eight-byte strings. I tested the techniques in both Altair 4.0 BASIC and BASIC-80, Version 5.02. They should also work with TRS-80 BASIC and other 8080-based Microsoft BASICs. The listed program was written in BASIC-80, running under CP/M on an Informer III computer from Advanced Informatics.

In business programming, you usually encounter only integers and numbers with two or three significant decimal places. These decimal numbers can be converted to integers by multiplying by 100 or 1000 and taking the integer value of the result. This conversion is worthwhile because integer values can be stored more compactly than floating point values.

You can pack integers between −32768 and +32767 in a two-byte string using the MKI$ function and unpack them using the CVI function. Integers between −8388608 (-2^{23}) and +8388607 ($2^{23}-1$) can be packed in a three-byte string, and integers between −2147483648 (-2^{31}) and +2147483647 ($2^{31}-1$) in a four-byte string. Unfortunately, there are no built-in functions to do the packing and unpacking for these integers.

The MKS$ function does pack a single-precision number into a four-byte string. All eight bits of one of these bytes are used to store an exponent, which allows floating point numbers between, roughly, 1E−38 and 1E+38 to be packed into the string. The remaining three bytes are used to store the mantissa. This restricts single-precision values to six digits of accuracy. Thus, numbers such as 1234567 stored as single-preci-

```
RUN
DATAPACK  -  03/10/80

DEMONSTRATES TWO METHODS OF PACKING DOUBLE PRECISION NUMBERS
INTO STRINGS LESS THAN EIGHT BYTES LONG.

THIS ALLOWS MORE EFFICIENT DATA STORAGE IN RANDOM BUFFERS.


NUMBER OF DECIMAL PLACES TO KEEP = ? 2

NUMBER TO BE PACKED = ? -12.345

-12.345 CAN BE PACKED INTO 2 BYTES, AS FOLLOWS:
00101101 11111011
ITS VALUE AFTER UNPACKING IS -12.35

NUMBER TO BE PACKED = ? 123.456

 123.456 CAN BE PACKED INTO 2 BYTES, AS FOLLOWS:
00111010 00110000
ITS VALUE AFTER UNPACKING IS  123.46

NUMBER TO BE PACKED = ? 12345.67

 12345.67 CAN BE PACKED INTO 3 BYTES, AS FOLLOWS:
10000111 11010110 10010010
ITS VALUE AFTER UNPACKING IS  12345.67

NUMBER TO BE PACKED = ? 1234567.89

 1234567.89 CAN BE PACKED INTO 4 BYTES, AS FOLLOWS:
00010101 11001101 01011011 10000111
ITS VALUE AFTER UNPACKING IS  1234567.89

NUMBER TO BE PACKED = ? 21474836.47

 21474836.47 CAN BE PACKED INTO 4 BYTES, AS FOLLOWS:
11111111 11111111 11111111 11111111
ITS VALUE AFTER UNPACKING IS  21474836.47

NUMBER TO BE PACKED = ? 9876543210.123

 9876543210.123 CAN BE PACKED INTO 6 BYTES, AS FOLLOWS:
11100010 11001100 10110110 10100000 10001010 10001100
ITS VALUE AFTER UNPACKING IS  9876543210.12
```

Sample run.

sion values are rounded off to six significant digits (CVS(MKS$(1234567)) = 1.23457E+06).

The upshot is that a number such as 1234567, which would fit in three bytes, must be stored in an eight-byte string using the MKD$ function. Otherwise, you sacrifice accuracy.

The Datapack Program

The listed program, called Datapack, includes user-defined functions to accurately pack 6.8 digits of accuracy into three-byte strings and 9.2 digits into four-byte strings. There is also a subroutine that will pack 10, 12 or 14 digits into five-, six- or seven-byte strings. The functions needed to unpack the resultant strings are also included.

The program first asks for the number of significant decimal places. This can be any nonzero integer, but zero, two or three is the most likely response. Numbers with more than the specified number of decimal places are rounded off while being converted to integers in the packing functions.

The main program loop asks for a number to be packed. It then tests the absolute value of the number and packs it into the smallest string into which it will fit, using one of the three packing functions or the one packing subroutine. The packed string, X$, is then unpacked into the variable X#.

The program then prints out the length and binary contents of the packed string, followed by the value of the unpacked string. The only difference between this value and the one submitted for packing is that it is rounded to the number of significant decimal places. Finally, the program branches back to the start of the loop and asks for another number to be packed.

Some Notes on the Program

Lines 130 through 200 contain the packing and unpacking functions for integers less than 2^{15}. The packing function, FNP2$, includes the ABS and SGN functions to ensure that negative fractions will be properly rounded. Otherwise, −10.445 would round off to −10.44 instead of −10.45 (assuming two decimal places are being kept).

The unpacking function, FNU2#, includes the VAL and STR$ functions to convert the single-precision result of CVI(I$)/DD into double precision. A function called CDBL is supposed to do this, but it doesn't work. (It didn't work in Altair BASIC either. The two companion functions, CSNG and CINT, don't do anything, so it is difficult to determine if they work. It is a wonder that they continue to document these functions at all.)

Lines 230 through 300 contain the packing and unpacking functions for integers less than 2^{23}. They make use of the fact that for values of D# between N3# -2^{23} and N3# $+2^{23}$, the strings returned by MKD$(D#) differ only in three of their eight bytes. It is only necessary to save these three significant bytes. The eight-byte string can be reconstructed before unpacking with CVD, as is done in the function FNU3#.

The functions FNP4$ and FNU4# are similar to FNP3$ and FNU3#. They use a different magic number, N4#, which works for integers less then 2^{31}. I first saw the number N4# used in the general ledger programs distributed by the Altair Computer Center. I found the number N3# by analysis, trial and error. There is probably a number N5# that would work for larger integers, but I have not looked for it.

The subroutine beginning at line 600 can be used to pack integers too big to fit into a four-byte string. As a subroutine, it is less convenient to work with than the packing functions. It also takes much longer to execute. It packs two digits of the input variable, D#, into each byte of the output string, X$. The high-order bit of each byte in X$ is also set to one if D# is positive.■

```
10 PRINT"DATAPACK  -  03/10/80"
20 PRINT
30 PRINT"DEMONSTRATES TWO METHODS OF PACKING DOUBLE PRECISION NUMBERS"
40 PRINT"INTO STRINGS LESS THAN EIGHT BYTES LONG. "
50 PRINT
60 PRINT"THIS ALLOWS MORE EFFICIENT DATA STORAGE IN RANDOM BUFFERS. "
70 PRINT: PRINT
80 '
90 INPUT"NUMBER OF DECIMAL PLACES TO KEEP = ";ND
100 DD=INT(10^ND)
110 '
120 '**********************************************************************
130 'THE FOLLOWING FUNCTION CONVERTS ITS ARGUMENT INTO AN INTEGER BETWEEN
140 '-32768 AND +32767 AND STORES THE RESULT IN A TWO BYTE STRING.
150 '
160 DEF FNP2$(D)=MKI$(SGN(D)*INT(ABS(D*DD)+.5))
170 '
180 'UNPACK THE RESULT OF THE ABOVE FUNCTION
190 '
200 DEF FNU2#(I$)=VAL(STR$(CVI(I$)/DD))
210 '**********************************************************************
220 '
230 'THE FOLLOWING FUNCTION CONVERTS ITS ARGUMENT INTO AN INTEGER BETWEEN
240 '-8388608 AND +8388607 AND STORES THE RESULT IN A THREE BYTE STRING.
250 '
260 N3#=549764202496#:DEF FNP3$(D#)=MID$(MKD$(N3#+D#*DD+.5),3,3)
270 '
280 'UNPACK THE RESULT OF THE ABOVE FUNCTION
290 '
300 DEF FNU3#(I$)=(CVD(CHR$(0)+CHR$(0)+I$+CHR$(0)+CHR$(0)+CHR$(168))-N3#)/DD
310 '**********************************************************************
320 '
330 'THE FOLLOWING FUNCTION CONVERTS ITS ARGUMENT INTO AN INTEGER BETWEEN
340 '-2147483648 AND +2147483647 AND STORES THE RESULT IN A FOUR BYTE STRING.
350 '
360 N4#=551903297536#:DEF FNP4$(D#)=MID$(MKD$(N4#+D#*DD+.5),3,4)
370 '
380 'UNPACK THE RESULT OF THE ABOVE FUNCTION
390 '
400 DEF FNU4#(I$)=(CVD(CHR$(0)+CHR$(0)+I$+CHR$(0)+CHR$(168))-N4#)/DD
410 '*********************************************************************
420 '
430 '
440 PRINT
450 INPUT"NUMBER TO BE PACKED = ";D#
460 '
470 IF ABS(D#*DD+.5)<32768!      THEN X$=FNP2$(D#): X#=FNU2#(X$): GOTO 520
480 IF ABS(D#*DD+.5)<8388608#    THEN X$=FNP3$(D#): X#=FNU3#(X$): GOTO 520
490 IF ABS(D#*DD+.5)<2147483648# THEN X$=FNP4$(D#): X#=FNU4#(X$): GOTO 520
500 GOSUB 630: GOSUB 690
510 '
520 PRINT
530 PRINT D#"CAN BE PACKED INTO"LEN(X$)"BYTES, AS FOLLOWS: "
540 GOSUB 760
550 PRINT"ITS VALUE AFTER UNPACKING IS "X#
560 GOTO 440
570 '
580 '*********************************************************************
590 '
600 'THE FOLLOWING SUBROUTINE PACKS AN INTEGER OF X DIGITS INTO A STRING
610 'OF LENGTH (X+1)\2.
620 '
630 J$=STR$(SGN(D#)*INT(ABS(D#*DD)+.5)):X$=""
640 FOR I=LEN(J$)-1 TO 1 STEP-2:X$=CHR$(ABS(VAL(MID$(J$,I,2)))-128*(D#>0))+X$
650 NEXT: RETURN
660 '
670 'UNPACK THE RESULT OF THE PREVIOUS SUBROUTINE
680 '
690 X#=0:IF ASC(X$)>127 THEN J#=1 ELSE J#=-1
700 FOR I=LEN(X$) TO 1 STEP -1:X#=X#+(ASC(MID$(X$,I,1))AND 127)*J#:J#=J#*100
710 :NEXT:X#=X#/DD:RETURN
720 '*********************************************************************
730 '
740 'DISPLAY BINARY CONTENTS OF PACKED NUMERIC STRINGS
750 '
760 FOR I=1 TO LEN(X$): BYTE=ASC(MID$(X$,I,1)): GOSUB 770: NEXT: PRINT: RETURN
770 FOR BIT=7 TO 0 STEP -1: K=2^BIT
780 IF K AND BYTE THEN PRINT"1"; ELSE PRINT"0";
790 NEXT: PRINT" ";: RETURN
```

Program listing. Datapack program in BASIC-80.

KILOBAUD KLASSROOM NO. 22

Machine-Language Programming

This month's assignment is to read this chapter on machine-language programming. You will find your homework assignment towards the end of the chapter. There will be a quiz. Be prepared next time for a discussion on assembly-language programming.

Peter A. Stark
PO Box 209
Mt. Kisco, NY 10549

This month I'll dive right into programming from a beginner's point of view (though I assume that everybody knows at least a bit of programming in BASIC).

Addresses vs Contents

Like most eight-bit microprocessors, the 6802 used in our Kilobaud Klassroom Komputer uses eight-bit data and 16-bit addresses. Thus, I could talk of eight-bit binary data, and 16-bit binary addresses, but as you have already learned, most discussion of such binary data uses either octal or hexadecimal numbers rather than binary.

Most modern programmers use hexadecimal rather than octal numbers, as I will when discussing the 6802. Since each hexadecimal digit (ranging from 0 through F) represents four binary bits, I use two hex digits to represent eight-bit data bytes, and four hex digits for 16-bit addresses. Thus, if you see a column of numbers such as

```
1000  4F
1001  8B
1002  41
1003  BD
```

you can safely assume that the four-digit numbers on the left represent addresses, while the two-digit numbers in the right column stand for eight-bit data.

In fact, such notation is often used to show the contents of computer memory. Since computer memory generally has thousands of separate locations, any time you list the contents of memory you must specify where in memory each number is located. In the above instance, the left column shows the address of a location, while the right column shows the contents of that location.

Beginners often have trouble with the notion that a specific memory location can have both an address and different contents. But you can think of memory as being divided into thousands of little post office boxes, each of which has a box number called an address as well as space for some contents (mail).

Just as adjacent boxes in the post office have consecutive box numbers, so adjacent locations in the memory have consecutive addresses. In a computer having 16-bit addresses, these addresses are numbered from 0000 to FFFF. When you list the contents of several memory locations, you generally list adjacent (consecutively numbered) locations, as in the above example.

The contents of memory, however, are different from the contents of post office boxes. For one thing, the PO box can hold several pieces of mail at once. Each location of computer memory, on the other hand, can hold only one number at a time. In fact, each time you put a new number into a particular memory location, the old number there is automatically erased.

Furthermore, a PO box can be empty, whereas a memory location must always hold some number, even if that number is garbage and not used. When you first turn on power, each memory location acquires some (useless) number, and holds that number until it is replaced by some other number in the course of using the computer.

A third difference is that taking a number out of a memory location (such as transferring it into another location) does not really remove that number from its original location. Rather than moving a number from one location to another, you are simply copying it, so that this same number is now in two places at the same time. In order to remove a number from some location, you must erase it. But erasing memory usually means simply replacing its contents with some new number (such as 00), not really leaving the location blank.

So when you see a table such as

```
1000  4F
1001  8B
1002  41
1003  BD
```

you are simply saying that location 1000 holds the number 4F, and so on. You are also assuming that each of the other thousands of memory locations holds some number, even though it is not listed here.

Memory Contents

If each memory location then holds an eight-bit number (byte), what is that number used for? In general, the contents of a given location could be any one of four things:

1. Garbage. If a location is not being used, then it may still have some number left over from a previous program, or from the time the computer was first powered up.

2. A numeric value. That is, that location could be used to hold the value of some constant or variable being used in a program. In many cases, constants and variables are spread out over several adjacent locations, and a particular eight-bit number could be just part of such a number.

3. One character of a string. Alphanumeric strings are generally stored in memory using the ASCII code, one character to a memory location.

4. A machine-language instruction, part of some program.

If you just look at the contents of memory, how can you tell what is a number, an ASCII character, an instruction or garbage?

If you look at just one specific location, you usually cannot tell at all what its content is. On the other hand, if you look at a group of adjacent locations, you can often get at least a fairly good idea from the context.

For example, if you see that a set of consecutive locations has the hex numbers 52 45 41 44 59, a knowledgeable programmer may recognize the ASCII codes for the letters R E A D Y. This is obviously a string—the chance of this sequence of numbers being a numeric value or some instruction is just too small. But you still don't know whether this is a useful string, or whether it is some garbage left over from a program run long ago.

Thus the thing to remember is that com-

		ADDRESSING MODES															BOOLEAN/ARITHMETIC OPERATION	COND. CODE REG.					
		IMMED			DIRECT			INDEX			EXTND			IMPLIED			(All register labels refer to contents)	5	4	3	2	1	0
OPERATIONS	MNEMONIC	OP	~	#	OP	~	#	OP	~	#	OP	~	#	OP	~	#		H	I	N	Z	V	C
Add	ADDA	8B	2	2	9B	3	2	AB	5	2	BB	4	3				A + M → A	↕	•	↕	↕	↕	↕
	ADDB	CB	2	2	DB	3	2	EB	5	2	FB	4	3				B + M → B	↕	•	↕	↕	↕	↕
Add Acmltrs	ABA													1B	2	1	A + B → A	↕	•	↕	↕	↕	↕
Add with Carry	ADCA	89	2	2	99	3	2	A9	5	2	B9	4	3				A + M + C → A	↕	•	↕	↕	↕	↕
	ADCB	C9	2	2	D9	3	2	E9	5	2	F9	4	3				B + M + C → B	↕	•	↕	↕	↕	↕
And	ANDA	84	2	2	94	3	2	A4	5	2	B4	4	3				A · M → A	•	•	↕	↕	R	•
	ANDB	C4	2	2	D4	3	2	E4	5	2	F4	4	3				B · M → B	•	•	↕	↕	R	•
Bit Test	BITA	85	2	2	95	3	2	A5	5	2	B5	4	3				A · M	•	•	↕	↕	R	•
	BITB	C5	2	2	D5	3	2	E5	5	2	F5	4	3				B · M	•	•	↕	↕	R	•
Clear	CLR							6F	7	2	7F	6	3				00 → M	•	•	R	S	R	R
	CLRA													4F	2	1	00 → A	•	•	R	S	R	R
	CLRB													5F	2	1	00 → B	•	•	R	S	R	R
Compare	CMPA	81	2	2	91	3	2	A1	5	2	B1	4	3				A − M	•	•	↕	↕	↕	↕
	CMPB	C1	2	2	D1	3	2	E1	5	2	F1	4	3				B − M	•	•	↕	↕	↕	↕
Compare Acmltrs	CBA													11	2	1	A − B	•	•	↕	↕	↕	↕
Complement, 1's	COM							63	7	2	73	6	3				$\overline{M}$ → M	•	•	↕	↕	R	S
	COMA													43	2	1	$\overline{A}$ → A	•	•	↕	↕	R	S
	COMB													53	2	1	$\overline{B}$ → B	•	•	↕	↕	R	S
Complement, 2's	NEG							60	7	2	70	6	3				00 − M → M	•	•	↕	↕	①	②
(Negate)	NEGA													40	2	1	00 − A → A	•	•	↕	↕	①	②
	NEGB													50	2	1	00 − B → B	•	•	↕	↕	①	②
Decimal Adjust, A	DAA													19	2	1	Converts Binary Add. of BCD Characters into BCD Format	•	•	↕	↕	↕	③
Decrement	DEC							6A	7	2	7A	6	3				M − 1 → M	•	•	↕	↕	4	•
	DECA													4A	2	1	A − 1 → A	•	•	↕	↕	4	•
	DECB													5A	2	1	B − 1 → B	•	•	↕	↕	4	•
Exclusive OR	EORA	88	2	2	98	3	2	A8	5	2	B8	4	3				A⊕M → A	•	•	↕	↕	R	•
	EORB	C8	2	2	D8	3	2	E8	5	2	F8	4	3				B⊕M → B	•	•	↕	↕	R	•
Increment	INC							6C	7	2	7C	6	3				M + 1 → M	•	•	↕	↕	⑤	•
	INCA													4C	2	1	A + 1 → A	•	•	↕	↕	⑤	•
	INCB													5C	2	1	B + 1 → B	•	•	↕	↕	⑤	•
Load Acmltr	LDAA	86	2	2	96	3	2	A6	5	2	B6	4	3				M → A	•	•	↕	↕	R	•
	LDAB	C6	2	2	D6	3	2	E6	5	2	F6	4	3				M → B	•	•	↕	↕	R	•
Or, Inclusive	ORAA	8A	2	2	9A	3	2	AA	5	2	BA	4	3				A + M → A	•	•	↕	↕	R	•
	ORAB	CA	2	2	DA	3	2	EA	5	2	FA	4	3				B + M → B	•	•	↕	↕	R	•
Push Data	PSHA													36	4	1	A → M_{SP}, SP − 1 → SP	•	•	•	•	•	•
	PSHB													37	4	1	B → M_{SP}, SP − 1 → SP	•	•	•	•	•	•
Pull Data	PULA													32	4	1	SP + 1 → SP, M_{SP} → A	•	•	•	•	•	•
	PULB													33	4	1	SP + 1 → SP, M_{SP} → B	•	•	•	•	•	•
Rotate Left	ROL							69	7	2	79	6	3				M	•	•	↕	↕	⑥	↕
	ROLA													49	2	1	A ← C ← b7 ← b0	•	•	↕	↕	⑥	↕
	ROLB													59	2	1	B	•	•	↕	↕	⑥	↕
Rotate Right	ROR							66	7	2	76	6	3				M	•	•	↕	↕	⑥	↕
	RORA													46	2	1	A C → b7 → b0	•	•	↕	↕	⑥	↕
	RORB													56	2	1	B	•	•	↕	↕	⑥	↕
Shift Left, Arithmetic	ASL							68	7	2	78	6	3				M	•	•	↕	↕	⑥	↕
	ASLA													48	2	1	A C ← b7 ← b0 ← 0	•	•	↕	↕	⑥	↕
	ASLB													58	2	1	B	•	•	↕	↕	⑥	↕
Shift Right, Arithmetic	ASR							67	7	2	77	6	3				M	•	•	↕	↕	⑥	↕
	ASRA													47	2	1	A b7 → b0 → C	•	•	↕	↕	⑥	↕
	ASRB													57	2	1	B	•	•	↕	↕	⑥	↕
Shift Right, Logic	LSR							64	7	2	74	6	3				M	•	•	R	↕	⑥	↕
	LSRA													44	2	1	A 0 → b7 → b0 → C	•	•	R	↕	⑥	↕
	LSRB													54	2	1	B	•	•	R	↕	⑥	↕
Store Acmltr.	STAA				97	4	2	A7	6	2	B7	5	3				A → M	•	•	↕	↕	R	•
	STAB				D7	4	2	E7	6	2	F7	5	3				B → M	•	•	↕	↕	R	•
Subtract	SUBA	80	2	2	90	3	2	A0	5	2	B0	4	3				A − M → A	•	•	↕	↕	↕	↕
	SUBB	C0	2	2	D0	3	2	E0	5	2	F0	4	3				B − M → B	•	•	↕	↕	↕	↕
Subtract Acmltrs.	SBA													10	2	1	A − B → A	•	•	↕	↕	↕	↕
Subtr. with Carry	SBCA	82	2	2	92	3	2	A2	5	2	B2	4	3				A − M − C → A	•	•	↕	↕	↕	↕
	SBCB	C2	2	2	D2	3	2	E2	5	2	F2	4	3				B − M − C → B	•	•	↕	↕	↕	↕
Transfer Acmltrs	TAB													16	2	1	A → B	•	•	↕	↕	R	•
	TBA													17	2	1	B → A	•	•	↕	↕	R	•
Test, Zero or Minus	TST							6D	7	2	7D	6	3				M − 00	•	•	↕	↕	R	R
	TSTA													4D	2	1	A − 00	•	•	↕	↕	R	R
	TSTB													5D	2	1	B − 00	•	•	↕	↕	R	R
																		H	I	N	Z	V	C

LEGEND:

OP	Operation Code (Hexadecimal);
~	Number of MPU Cycles;
#	Number of Program Bytes;
+	Arithmetic Plus;
−	Arithmetic Minus;
·	Boolean AND;
M_{SP}	Contents of memory location pointed to be Stack Pointer;
+	Boolean Inclusive OR;
⊕	Boolean Exclusive OR;
$\overline{M}$	Complement of M;
→	Transfer Into;
0	Bit = Zero;
00	Byte = Zero;

Note – Accumulator addressing mode instructions are included in the column for IMPLIED addressing

CONDITION CODE SYMBOLS:

H	Half-carry from bit 3;
I	Interrupt mask
N	Negative (sign bit)
Z	Zero (byte)
V	Overflow, 2's complement
C	Carry from bit 7
R	Reset Always
S	Set Always
↕	Test and set if true, cleared otherwise
•	Not Affected

Table 1. Accumulator and memory instructions. (All tables courtesy of Motorola.)

puter memory can contain any of the four types of contents, but generally like contents are grouped together.

But machine-language programs are like BASIC programs—some programmers will put BASIC's DATA statements at the very end of a program, while others may bury them between other statements. In a like way, machine-language programmers may

		IMMED			DIRECT			INDEX			EXTND			IMPLIED				COND. CODE REG. 5	4	3	2	1	0
POINTER OPERATIONS	MNEMONIC	OP	~	#	OP	~	#	OP	~	#	OP	~	#	OP	~	#	BOOLEAN/ARITHMETIC OPERATION	H	I	N	Z	V	C
Compare Index Reg	CPX	8C	3	3	9C	4	2	AC	6	2	BC	5	3				$X_H - M, X_L - (M+1)$	•	•	7	↕	8	•
Decrement Index Reg	DEX													09	4	1	$X - 1 \rightarrow X$	•	•	•	↕	•	•
Decrement Stack Pntr	DES													34	4	1	$SP - 1 \rightarrow SP$	•	•	•	•	•	•
Increment Index Reg	INX													08	4	1	$X + 1 \rightarrow X$	•	•	•	↕	•	•
Increment Stack Pntr	INS													31	4	1	$SP + 1 \rightarrow SP$	•	•	•	•	•	•
Load Index Reg	LDX	CE	3	3	DE	4	2	EE	6	2	FE	5	3				$M \rightarrow X_H, (M+1) \rightarrow X_L$	•	•	9	↕	R	•
Load Stack Pntr	LDS	8E	3	3	9E	4	2	AE	6	2	BE	5	3				$M \rightarrow SP_H, (M+1) \rightarrow SP_L$	•	•	9	↕	R	•
Store Index Reg	STX				DF	5	2	EF	7	2	FF	6	3				$X_H \rightarrow M, X_L \rightarrow (M+1)$	•	•	9	↕	R	•
Store Stack Pntr	STS				9F	5	2	AF	7	2	BF	6	3				$SP_H \rightarrow M, SP_L \rightarrow (M+1)$	•	•	9	↕	R	•
Indx Reg → Stack Pntr	TXS													35	4	1	$X - 1 \rightarrow SP$	•	•	•	•	•	•
Stack Pntr → Indx Reg	TSX													30	4	1	$SP + 1 \rightarrow X$	•	•	•	•	•	•

Table 2. Index register and stack manipulation instructions.

put all their numeric and string data together at the end of a program (or perhaps into a completely separate area altogether), while others intersperse it between other program instructions. But unlike BASIC, which recognizes DATA statements and simply ignores them in the middle of a program and jumps over them, a machine-language program must in some way have the equivalents of GOTO statements just before data to make sure the computer doesn't accidentally try to perform it as if it were instructions. Thus, you must always be aware of what is numeric or string data, and what is program.

There is another interesting difference between programming in BASIC and in machine language. In BASIC, you generally just type RUN, and the computer knows where to begin—usually the top line of the program (or the first line which is not a REM). But in machine-language programs, the program could lie anywhere in a very large memory, and the computer has no way of finding its first instruction unless you tell it. Thus, starting a machine-language program always involves specifying some starting address.

If, for example, you tell the computer to start executing a program at location 1000, it will perform the instruction at that address, and then proceed in consecutive addresses—1001, 1002, 1003, etc.—until it encounters some instruction similar to BASIC's GOTO, GOSUB, IF or perhaps STOP.

Unlike BASIC (which generally performs some error checking and refuses to perform obviously wrong instructions), the computer does no error checking when executing a machine-language program. If there is a wrong instruction, or perhaps numeric or string data or garbage, in the midst of a real program, the computer will continue through it, trying to execute it as if it were a real program. It simply cannot tell the difference.

Multi-Byte Instructions

A typical memory location can only hold an eight-bit byte, which can have one of 256 different values (hex 00 through FF, which corresponds to the decimal numbers 0 through 255). This is not enough of a range to represent a wide variety of different instructions. Thus, in most microprocessors, instructions may be spread out over more than one location.

In the 6802, instructions can consist of one, two or three bytes. Each particular instruction has a specific length; when we use that instruction we must use the correct number of bytes, and when the computer performs that instruction, it looks for that same number.

For example, here is a portion of a 6802 program:

```
1000  4F
1001  8B
1002  41
1003  BD
1004  E1
1005  D1
```

Although these six bytes occupy six locations, in reality there are only three instructions. Rather than write the program in this way, we generally write it as

```
1000  4F
1001  8B 41
1003  BD E1D1
```

which groups the bytes of each instruction together on one line. You see here a one-byte instruction (4F) in location 1000; a two-byte instruction (8B 41), which starts at location 1001; and a three-byte instruction (BD E1D1), which starts at location 1003.

OPERATIONS	MNEMONIC	RELATIVE OP	RELATIVE ~	RELATIVE #	INDEX OP	INDEX ~	INDEX #	EXTND OP	EXTND ~	EXTND #	IMPLIED OP	IMPLIED ~	IMPLIED #	BRANCH TEST	COND. CODE REG. 5 H	4 I	3 N	2 Z	1 V	0 C
Branch Always	BRA	20	4	2										None	•	•	•	•	•	•
Branch If Carry Clear	BCC	24	4	2										C = 0	•	•	•	•	•	•
Branch If Carry Set	BCS	25	4	2										C = 1	•	•	•	•	•	•
Branch If = Zero	BEQ	27	4	2										Z = 1	•	•	•	•	•	•
Branch If $\geq$ Zero	BGE	2C	4	2										$N \oplus V = 0$	•	•	•	•	•	•
Branch If $>$ Zero	BGT	2E	4	2										$Z + (N \oplus V) = 0$	•	•	•	•	•	•
Branch If Higher	BHI	22	4	2										$C + Z = 0$	•	•	•	•	•	•
Branch If $\leq$ Zero	BLE	2F	4	2										$Z + (N \oplus V) = 1$	•	•	•	•	•	•
Branch If Lower Or Same	BLS	23	4	2										$C + Z = 1$	•	•	•	•	•	•
Branch If $<$ Zero	BLT	2D	4	2										$N \oplus V = 1$	•	•	•	•	•	•
Branch If Minus	BMI	2B	4	2										N = 1	•	•	•	•	•	•
Branch If Not Equal Zero	BNE	26	4	2										Z = 0	•	•	•	•	•	•
Branch If Overflow Clear	BVC	28	4	2										V = 0	•	•	•	•	•	•
Branch If Overflow Set	BVS	29	4	2										V = 1	•	•	•	•	•	•
Branch If Plus	BPL	2A	4	2										N = 0	•	•	•	•	•	•
Branch To Subroutine	BSR	8D	8	2										See Special Operations	•	•	•	•	•	•
Jump	JMP				6E	4	2	7E	3	3					•	•	•	•	•	•
Jump To Subroutine	JSR				AD	8	2	BD	9	3					•	•	•	•	•	•
No Operation	NOP										01	2	1	Advances Prog. Cntr. Only	•	•	•	•	•	•
Return From Interrupt	RTI										3B	10	1		10					
Return From Subroutine	RTS										39	5	1	See Special Operations	•	•	•	•	•	•
Software Interrupt	SWI										3F	12	1		•	•	•	•	•	•
Wait for Interrupt*	WAI										3E	9	1		•	11	•	•	•	•

*WAI puts Address Bus, R/W, and Data Bus in the three-state mode while VMA is held low.

Table 3. Jump and branch instructions.

Notice that multi-byte instructions take up several locations, but only the address of the starting location is shown. Since the second instruction has an address of 1001, while the following one is shown at 1003, you can conclude that the number 41 must have been in location 1002.

Although machine-language instructions are different lengths, the first byte (such as 4F, 8B or BD in the above example) has a special meaning. It is called the instruction code, operation code or often just op code, because it is a coded number that specifies exactly the kind of operation to be performed.

(Note the similarity to BASIC, where the first word on a statement—such as REM, LET, IF, READ, etc.—is a keyword which specifies exactly what that statement is to do.)

In the 6800 and 6802 processors, op codes are always one byte long; in some other processors they may sometimes be longer.

Some operations are completely specified by the op code alone, and need no other information. For example, the 4F code in the first instruction above tells the 6802 to clear the A accumulator to 00. This is specific enough to require no further details.

Other operations (such as the second and third instructions above) require additional information which is then represented by one or two additional bytes which are called the operand. For example, in the second instruction above, the 8B is the op code while 41 is the operand.

In general, every op code describes a specific operation and requires a specific instruction length. For example, the op code 4F in the 6802 is always a one-byte instruction, while 8B is always followed by exactly one more byte to specify the operand.

The meaning of every possible op code is defined by the manufacturer of the microprocessor, and is part of what is called the instruction set which is then published in the processor spec sheets.

The 6802 Instruction Set

For the 6802, Tables 1 through 4 (reprinted through the courtesy of Motorola) list every possible instruction in a concise way. *Don't panic!* Although these tables look formidable, they contain a lot more information than we usually need. Moreover, if you use assembly language and an assembler, you need not even refer to these tables in most instances. In fact, many programmers only use a small subset of all these instructions for the simple reason that they don't even remember that the others exist.

These four tables break up the 6802 instruction set into different groups:

Table 1 describes those instructions used to manipulate memory and the two accumulators. As you can see by looking at the left-hand column of the table, these instructions allow you to add and subtract, clear memory or accumulator contents, do comparisons and other such numeric operations.

Table 2 lists those instructions affecting the index and stack pointer registers. With these instructions you can increment or decrement these registers, load and save their contents in memory or interchange their contents.

Table 3 lists jump and branch instructions. These instructions are similar to the GOTO, GOSUB, IF and RETURN instructions of BASIC, except that instead of just one IF statement, machine language has about a dozen different forms of "branch if . . . " instructions.

Finally, Table 4 lists those instructions affecting just the six-bit condition code register.

In the three-step sample program above, the second instruction had the op code 8B. This op code is at the very beginning of Table 1. Let's look at the top left corner of this table a bit closer. It looks like this:

OPERATIONS	MNEMONIC	IMMED OP	IMMED ~	IMMED #
Add	ADDA	8B	2	2

The left column tells you that this op code is used to do an addition; in fact, the notation A + M→A in one of the right-hand columns tells you that the number in accumulator A is added to a number in memory, and the result goes back into accumulator A.

The second column tells us that the mnemonic for this instruction is ADDA, meaning "ADD to A." The mnemonic is a three- or four-letter code that programmers use to remind them of the function of instructions so they don't have to remember their numeric op codes; it is certainly easier to remember what ADDA means than what 8B means. (When I get to assembly language, you will see that assembly programs are written with mnemonics, and the assembler automatically translates from mnemonics to the actual numeric op codes. Thus you do not have to know or memorize the op codes themselves.) Although the mnemonic here is given as ADDA, it is common to write it as ADD A to separate the words from each

		IMPLIED				COND. CODE REG.					
						5	4	3	2	1	0
OPERATIONS	MNEMONIC	OP	~	#	BOOLEAN OPERATION	H	I	N	Z	V	C
Clear Carry	CLC	0C	2	1	0 → C	•	•	•	•	•	R
Clear Interrupt Mask	CLI	0E	2	1	0 → I	•	R	•	•	•	•
Clear Overflow	CLV	0A	2	1	0 → V	•	•	•	•	R	•
Set Carry	SEC	0D	2	1	1 → C	•	•	•	•	•	S
Set Interrupt Mask	SEI	0F	2	1	1 → I	•	S	•	•	•	•
Set Overflow	SEV	0B	2	1	1 → V	•	•	•	•	S	•
Acmltr A → CCR	TAP	06	2	1	A → CCR	12					
CCR → Acmltr A	TPA	07	2	1	CCR → A	•	•	•	•	•	•

CONDITION CODE REGISTER NOTES: (Bit set if test is true and cleared otherwise)

1 (Bit V) Test: Result = 10000000?
2 (Bit C) Test: Result = 00000000?
3 (Bit C) Test: Decimal value of most significant BCD Character greater than nine? (Not cleared if previously set.)
4 (Bit V) Test: Operand = 10000000 prior to execution?
5 (Bit V) Test: Operand = 01111111 prior to execution?
6 (Bit V) Test: Set equal to result of N⊕C after shift has occurred.
7 (Bit N) Test: Sign bit of most significant (MS) byte = 1?
8 (Bit V) Test: 2's complement overflow from subtraction of MS bytes?
9 (Bit N) Test: Result less than zero? (Bit 15 = 1)
10 (All) Load Condition Code Register from Stack. (See Special Operations)
11 (Bit I) Set when interrupt occurs. If previously set, a Non-Maskable Interrupt is required to exit the wait state.
12 (All) Set according to the contents of Accumulator A.

Table 4. Condition code register manipulation instructions.

other.

Most 6802 instructions have several different forms, depending on how the operand is specified. The code 8B specifies a particular form of the ADDA instruction called immediate. (This is what is meant by IMMED in the table; the other forms of the ADDA instruction are the DIRECT, INDEXed, and EXTeNDed forms.)

In the table, you see an entry of 8B 2 2. 8B is the numeric op code (in the OP column). The 2 in the ~ column tells you that this particular instruction always takes exactly two machine clock cycles. Since in our Kilobaud Klassroom Komputer a clock cycle takes 1.11 microseconds, you can see that this ADDA instruction will always take 2.22 microseconds. The execution time is important to us only when we are writing a program which must execute in some precisely known time.

Finally, the 2 in the # column tells you that this instruction always has exactly two bytes (8B 41 in our example above).

As you can see, different forms of the ADDA run slightly differently. For example, the extended form is listed in Table 1 as BB 4 3, meaning that the op code is BB, that it requires four machine cycles to execute, and that it is a three-byte instruction. This appears to complicate the situation, but since few people program directly in machine language it is not as serious as it sounds. When you program in assembly language, the assembler automatically takes care of choosing the correct form of an instruction, and even using the correct number of bytes. Thus it is seldom necessary to consult the fine print in these tables.

(The six columns at the right, labelled HINZVC, refer to the condition code register; I'll cover those later.)

Addressing Modes

As you will note in Table 1, there are five columns labelled ADDRESSING MODES. (There is actually a sixth mode, used only for instructions in Table 3). These columns give the various forms of an instruction. Some instructions are available in only one mode; others may exist in several modes.

Let's look first at those instructions in Table 1 which only have the Implied Mode form; this includes instructions such as ABA (Add B accumulator to A), CLR A and CLR B (Clear A or B accumulator), CBA (Compare B with A) and so on. You will note that each of these op codes has # equal to 1, meaning that those instructions are single-byte instructions.

Implied mode instructions are the easiest to understand, because their function is very clear-cut. The op code is sufficiently explicit that no operand is required with it to give further details.

All other instructions listed in Table 1 require one or two additional operand bytes to give the computer some additional information.

Look, for example, at ADD A as compared with ABA. ABA says "Add B to A"; this is a complete description which implies that the number in accumulator B is added to the number in accumulator A, and the answer is left in A.

But ADD A leaves open the question of "add what?" In this case, some number from memory is to be added to accumulator A, and the result must be left in A; but the job of the operand bytes is to specify where the number in memory is located. Since there are four forms of the ADD A instruction, there are four different ways of specifying the location of the number to be added to A.

Immediate Mode—In the immediate mode, the number to be added is in the byte immediately after the op code. For instance, our sample program above had the instruction

```
1001  8B 41
```

which means "add the hex number 41 to accumulator A." Since the op code 8B is in location 1001, the computer looks in the very next location, at 1002, for the number to be added since the op code 8B always means that the number to be added is immediately after the op code.

In the same way, 8B 01 would mean "add a 1," while 8B FF would mean "add FF to accumulator A." Thus you see that in the im-

Program listing.

```
0100 PRINT "6800/6802 CROSS-ASSEMBLER IN BASIC"
0110 PRINT "COPYRIGHT 1980 BY PETER A. STARK"
0120 PRINT "ALL RIGHTS RESERVED"
0130 PRINT

0135 REM ON TRS-80 USE "CLEAR 100"

0140 DIM N$(200), L9(200)
0150 DIM D$(100), D8(100)
0160 D6 = 100
0170 LINE= 0                                    : REM NOT NEEDED ON MANY SYSTEMS
0180 L7 = 0
0190 L8 = 0
0200 L1 = 1
0210 P4 = 1

0220 REM ZERO OUT DEFER ADDRESSES
```

mediate mode, numbers to be added (or used in some other way) are placed directly into the instruction. All the immediate mode instructions in Table 1 are two bytes long—the first byte is always the op code, while the second byte is the number being used.

Extended Mode—As you will note from Table 1, all the extended mode instructions are three bytes long. In each case, the first byte is the op code, and the second and third bytes contain the address of the location in memory where the number being used is located.

For example, the extended form of the ADD A instruction has op code BB. Thus an instruction BB 3328 would mean that the computer should take a number out of location 3328, and add that number to accumulator A.

This is an important concept to understand. BB 3328 does not mean "add 3328"! Instead, it means "add the contents of location 3328." The difference between the immediate and extended forms of instructions is confusing at first, but essential. (Many older computers have only extended mode addressing.)

Either immediate or extended addressing can do the same job. For instance, if you want to add a 5 to accumulator A, you could do so with an immediate instruction simply by putting in an instruction which says 8B 05. Alternatively, you could put that 05 into some otherwise unused location (such as 7322, for instance) and then use the instruction BB 7322 to add the contents of location 7322, a 5, to the accumulator. Both of these methods would work, but the immediate form is shorter and quicker since it requires only two bytes (and two machine cycles) instead of three bytes (plus a fourth to hold the number 05, and four machine cycles). Thus having an immediate mode is very useful in microcomputers where saving space and time may be important.

Direct Mode—Direct mode is similar to extended mode, except that it is used specifically with addresses which start with 00. In the 6802, memory locations 0000 through 00FF are said to comprise the direct page; the direct mode is specifically intended for accessing data on this page. In direct instructions, the 00 part of the operand address is omitted. For example, the instructions

```
9B 4D
BB 004D
```

both add the number in location 004D to accumulator A. Direct mode saves one byte over extended mode, and also requires one less machine cycle to execute.

Because data placed on the direct page is easier and faster to get than data elsewhere in memory (via direct mode instructions), most 6802 users reserve this page for frequently used data so as to get the great-

```
0230 FOR I = 1 TO D6
0240      D8( I ) = -1
0250 NEXT I

0260 REM MAIN ASSEMBLER LOOP

0270 INPUT L$,C$,O$,R$
0280 X$ = "  "
0290 Y$ = "  "
0300 Z$ = "  "
0310 T$=""                                        : REM INDIRECT FLAG

0320 REM CHECK FOR COMMENT

0330 IF LEFT$(L$,1) <> "*" GOTO 380
0340 IF L1 = 0 GOTO 270
0350 PRINT #P4, TAB(16); L$; " "; C$; " "; O$; " "; R$
0360 GOTO 270

0370 REM CHECK FOR DUPLICATE LABEL

0380 IF L$="" GOTO 450
0390 A$ = L$
0400 GOSUB 3000                                   : REM FIND LABEL
0410 IF A8 < 0 GOTO 450
0420 PRINT "DUPLICATE LABEL", L$
0430 GOTO 270

0440 REM LIST DIRECTIVE?

0450 IF LEFT$(C$, 3) <>"LIS" GOTO 500
0460 L1 = 1
0470 GOSUB 2970                                   : REM PRINT AL
0480 GOTO 270

0490 REM NOLIST DIRECTIVE?

0500 IF LEFT$(C$,3) <> "NOL" GOTO 540
0510 L1 = 0
0520 GOTO 270
0530 REM END DIRECTIVE?

0540 IF C$ <> "END" GOTO 650
0550 GOSUB 2970                                   : REM PRINT AL

0560 REM SEARCH FOR UNDEFINED LABELS

0570 FOR I = 1 TO D6
0580      IF D8( I ) = -1 GOTO 620
0590      PRINT #P4, "UNDEFINED LABEL: ";
0600      L$ = MID$( D$( I ),4,6 )
0610      PRINT #P4, L$
0620 NEXT I
0630 END

0640 REM ON EACH NEW LABEL, SEARCH ARRAY FOR DEFERRED LABELS

0650 IF L$ = "" GOTO 990
0660 L4 = L8
0670 FOR I = 1 TO D6
0680      IF L$ <> MID$(D$( I ),4,6) GOTO 960

0690      REM FOUND ONE

0700      L3 = VAL(LEFT$(D$( I ),1))
0710      L8 = D8( I )
0720      X$ = MID$( D$( I ),2,2 )
0730      IF L3 = 3 GOTO 880

0740      REM LENGTH = 2

0750      O7 = L4 - D8( I ) - 2
0760      IF O7 >= -128 GOTO 800
0770      PRINT "BRANCH OUT OF RANGE"
0780      O7 = 0
0790      GOTO 820
0800      IF O7 > 127 GOTO 770
0810      IF O7 < 0 THEN O7 = O7+256
0820      B = O7
0830      GOSUB 3640                              : REM CONVERT 2 HEX
0840      Y$ = B$
0850      GOSUB 2910                              : REM GO PRINT ML ONLY
0860      GOTO 940                                : REM GO ERASE ENTRY

0870      REM LENGTH = 3

0880      A = L4
0890      GOSUB 3730                              : REM CONVERT 4 HEX
0900      Y$ = A$
0910      Z$ = ""
0920      GOSUB 2910                              : REM PRINT ML ONLY
0930      X$ = "  "
0940      Y$ = "  "
```

TIS-APL

- Stand alone **APL** for **Z80** includes OS & DOS.
- User work space exceeds 27K with some systems over 32K bytes.
- Use of system commands, file and system functions.
- 3D arrays inner and outer products.
- Catenate, scan, compress, reduce and rotate along specified axis.
- Custom versions for many popular Z80 based computers.
- Systems functions for:

  Communicating with other processors including large mainframe computers.

  Full ASC II and APL interface with wide variety of peripherals.

  Internal switching between APL and ASC II.

  Calls to User written assembler routines.

✓118

TELECOMPUTE INTEGRATED SYSTEMS INC.
251 SPADINA AVENUE, TORONTO
ONTARIO, CANADA M5T 2E2
PHONE: 416-363-9295

MICROSTAT NOW AVAILABLE FOR CP/M*

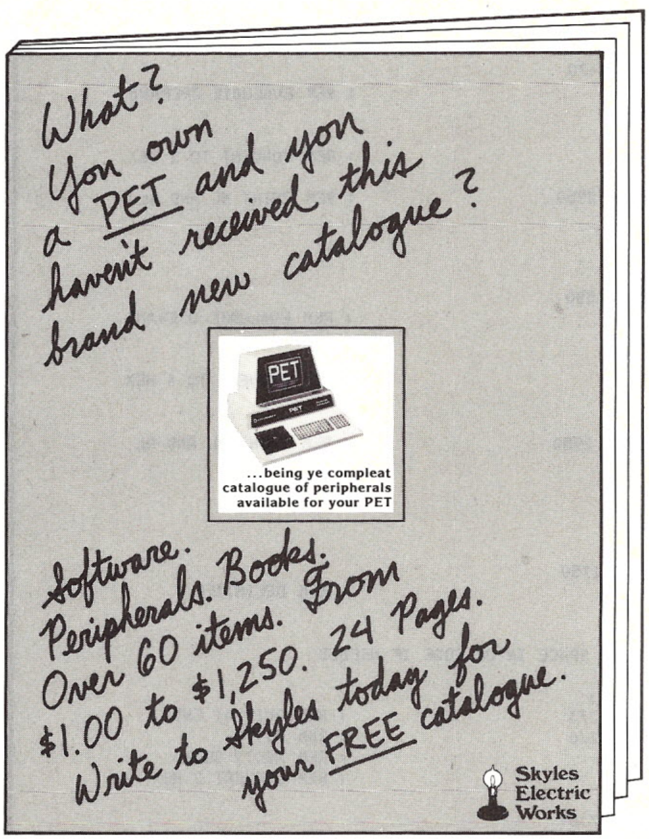
MICROSTAT, the most powerful statistics package available for microcomputers, is completely file-oriented with a powerful Data Management Subsystem (DMS) that allows you to edit, delete, augment, sort, rank-order, lag and transform (11 transformations, including linear, exponential and log) existing data into new data. After a file is created with DMS, Microstat provides statistical analysis in the following general areas: Descriptive Statistics (mean, sample, and population S.D., variance, etc.), Frequency Distributions (grouped or individual), Hypothesis Testing (mean or proportion), Correlation and Regression Analysis (with support statistics), Non-parametric Tests (Kolmogorov-Smirnov, Wilcoxon, etc.), Probability Distributions (8 of them), Crosstabs and Chi-square, ANOVA (one and two way), Factorials, Combinations and Permutations, plus other unique and useful features.

MICROSTAT requires 48K, Microsoft MBasic with CP/M and is sent on a single-density 8″ Disk. It is also available on 5″ diskettes for North Star DOS and Basic (32K and two drives recommended), specify which when ordering. The price for Microstat is $250.00. The user's manual is $15.00 and includes sample data and printouts. We have other business and educational software, call or write:

master charge THE INTERBANK CARD

ECOSOFT ✓82
P.O. Box 68602
Indianapolis, IN 46268
(317) 283-8883

*CP/M is a registered trade mark of Digital Research.



Skyles Electric Works
231 E South Whisman Road
Mountain View, CA 94041
✓66

Educational Software Professionals, Ltd.

APPLE

HI-RES TITRATION
HI-RES TITRATION is a simulation of an acid base titration using the Apple high-resolution color graphics. The program was written for use as a prelaboratory preparation and practice. It is effective as a lecture demonstration or for use by individual students. The program was written for high school use, but it is also useful for college chemistry classes.
32K Applesoft with Disk $19.95

RETAILING MATH
RETAILING MATH is a computer assisted instruction in the fundamentals of pricing mathematics. Lessons and reviews dealing with the mark-up equation, mark-up percentage and determining retail and cost prices are reinforced thru interactive exercises after each lesson by vocationally certified instructions.
48K Applesoft with Disk $39.95

EDUCATIONAL CHARADES
EDUCATIONAL CHARADES is an age-long game modified for classroom use. The teacher can create charade files relative to the current subject being taught.
32K Applesoft with Disk $19.95

APPLE-GRAMMER requires 32K in Applesoft and a disk drive $19.95

TEACHERS' AIDE I with Multi-Choice format requires 32K in Applesoft and a disk drive 19.95

TEACHERS' AIDE II with True and False format requires 32K in Applesoft and a disk drive $19.95

38437 Grand River • Farmington Hills, MI 48018
(313) 477-4470 ✓156
Dealer Inquiries Welcome

est time and space saving from using direct mode instructions.

Indexed Mode—This mode is perhaps the hardest for the beginner to understand. Like direct mode instructions, indexed instructions are two bytes long. But the second byte, rather than referring to a location on the direct page, is instead added to the contents of the index register to get the actual effective address being used.

For example, suppose the instruction AB 4D is executed at a time when the index register contains the number 4000. The processor adds the 4D to the 4000 from the index register to get 404D, and then uses the contents of location 404D in the instruction.

Table 3 shows instructions which use still another addressing mode, the relative mode. Relative mode instructions perform functions similar to BASIC's GOTO and IF . . . GOTO instructions.

In relative mode instructions, the computer takes the second byte of the instruction, treats it as a two's complement number, and adds it to the address of the next instruction to compute the address to which it should execute the GOTO.

For example, the op code 20 stands for BRAnch, which is the equivalent of a GOTO. The instruction 20 05 always means "GOTO the instruction five locations past the next instruction after the current one." While this seems quite difficult to follow, let's leave it at that for the moment and simply say that if we use assembly language and an assembler, we don't have to concern ourselves with the fine points since the assembler figures out the correct addresses automatically.

Some Simple Examples

Suppose, for instance, that you want to add two numbers and store the sum into location 2000. One way would be to use the following program:

```
1000  4F        Clear accumulator A
1001  8B 05     Add 5 to A
1003  8B 02     Add a 2 to get 7
1005  B7 2000   Store result in location 2000.
1008  7E 1008   GOTO 1008
```

At the end I used a 7E instruction, an extended instruction listed in Table 3, which stands for JMP or Jump—identical to BASIC's GOTO. Notice that the instruction

```
1008  GOTO 1008
```

in BASIC would tie up the program in an infinite loop. It does the same here. As you remember, a few pages ago I mentioned that if you do not tell the computer to stop at the end of a program, it will continue through memory, pulling out garbage from memory and trying to execute it as if it were a program. To make sure that the program stops at the end, you must put in some kind of a stop. Stopping in an infinite loop seems as good a way as any to keep from going on.

(You don't have to worry about this in BA-

```
0950      D8(I) = -1
0960 NEXT I
0970 L8 = L4

0980 REM RMB DIRECTIVE?
0990 IF C$ <> "RMB" GOTO 1160
1000 GOSUB 3090                                   : REM SAVE LABEL
1010 GOSUB 3150                                   : REM EVALUATE OPERAND
1020 IF O7 < 0 GOTO 1110
1030 A = L8
1040 L8 = L8 + O7
1050 GOSUB 3730                                   : REM CONVERT TO 4 HEX
1060 IF L1 = 0 GOTO 270
1070 PRINT #P4, "     ("; A$; ")";                : REM OTHERWISE PRINT
1080 GOSUB 2970                                   : REM PRINT AL
1090 GOTO 270

1100 REM ERROR MESSAGES

1110 PRINT "INVALID OPERAND"
1120 GOTO 1080
1130 PRINT "INVALID OPERATION CODE"
1140 GOTO 1080

1150 REM ORG DIRECTIVE?

1160 IF C$ <> "ORG" GOTO 1230
1170 GOSUB 3150                                   : REM EVALUATE OPERAND
1180 IF O7<0 GOTO 1110
1190 L8 = O7
1200 A = L8
1210 GOTO 1050

1220 REM EQU DIRECTIVE?

1230 IF C$ <> "EQU" GOTO 1350
1240 GOSUB 3150                                   : REM EVALUATE OPERAND
1250 IF O7<0 GOTO 1110
1260 IF L$ <> "" GOTO 1300
1270 PRINT "ERROR - MISSING LABEL"
1280 GOSUB 2970                                   : REM PRINT AL
1290 GOTO 270
1300 L7 = L7 + 1
1310 N$(L7) = L$
1320 L9(L7) = O7
1330 A = O7
1340 GOTO 1050
1350 GOSUB 3090                                   : REM SAVE LABEL

1360 REM FCB DIRECTIVE?

1370 IF C$ <> "FCB" GOTO 1470
1380 GOSUB 3150                                   : REM EVALUATE OPERAND
1390 IF O7 < 0 GOTO 1110
1400 B = O7
1410 GOSUB 3640                                   : REM CONVERT TO 2 HEX
1420 Y$ = B$
1430 IF L1 = 1 THEN GOSUB 2950                    : REM PRINT ML AND AL
1440 L8 = L8 + 1
1450 GOTO 270

1460 REM FDB DIRECTIVE?
1470 IF C$ <> "FDB" GOTO 1580
1480 GOSUB 3150                                   : REM EVALUATE OPERAND
1490 IF O7<0 GOTO 1110
1500 A = O7
1510 GOSUB 3730                                   : REM CONVERT TO 4 HEX
1520 Y$ = A$
1530 Z$ = ""
1540 IF L1 = 1 THEN GOSUB 2950                    : REM PRINT ML AND AL
1550 L8 = L8 + 2
1560 GOTO 270

1570 REM FCC DIRECTIVE?

1580 IF C$ <> "FCC" GOTO 1750
1590 D$ = LEFT$(O$,1)                             : REM DELIMITER
1600 FOR I=2 TO 32

1610      REM FIRST DELETE SPACE IN OP CODE IF NEEDED

1620      A$ = MID$(O$,I,1)
1630      IF A$ = "" GOTO 270                     : REM QUIT AT END
1640      IF A$ = D$ GOTO 270                     : REM DITTO
1650      B = ASC(A$)                             : REM ASCII CODE
1660      GOSUB 3640                              : REM CONVERT 2 HEX
1670      Y$=B$
1680      IF L1 = 0 GOTO 1710
1690      IF I = 2 THEN GOSUB 2950                : REM PRINT ML AND AL
1700      IF I <> 2 THEN GOSUB 2910               : REM PRINT ML ONLY
1710      L8 = L8 + 1
1720 NEXT I
1730 GOTO 270
```

```
1740 REM EXECUTABLE INSTRUCTION

1750 IF LEN(C$) <> 5 GOTO 1790
1760 IF MID$(C$,4,1) <> " " GOTO 1790
1770 C$ = LEFT$(C$,3)+ RIGHT$(C$,1)

1780 REM IS IT INDEXED?

1790 IF LEFT$(R$,1) <> "X" GOTO 1970
1800 GOSUB 3150                          : REM EVALUATE OPERAND
1810 R$=""                               : REM NO COMMENTS ALLOWED
1820 O$ = O$ + ",X"                      : REM INSERT THE ,X
1830 IF O7<0 GOTO 1110
1840 IF O7>255 GOTO 1110
1850 S$ = C$ + "X"
1860 GOSUB 3820                          : REM FIND OP CODE
1870 B = O7
1880 GOSUB 3640                          : REM CONVERT 2 HEX
1890 Y$ = B$
1900 B = O6
1910 GOSUB 3640                          : REM CONVERT 2 HEX
1920 X$ = B$
1930 IF L1 = 1 THEN GOSUB 2950           : REM PRINT ML AND AL
1940 L8 = L8 + 2
1950 GOTO 270
1960 REM IS IT IMMEDIATE?

1970 IF LEFT$(O$,1) <> "#" GOTO 2130
1980 T$="#"
1990 S$ = C$ + "#"
2000 L = LEN(O$)
2010 O$ = RIGHT$(O$,L-1)
2020 GOSUB 3150                          : REM EVALUATE OPERAND
2030 GOSUB 3820                          : REM FIND OPCODE

2040 REM SEPARATE TWO-BYTE OPERANDS

2050 IF C$ = "LDX" GOTO 2600
2060 IF C$ = "CPX" GOTO 2600
2070 IF C$ = "LDS" GOTO 2600

2080 REM ONE-BYTE IMMEDIATE INSTRUCTIONS

2090 IF O7>255 GOTO 1110
2100 IF O7<0 GOTO 1110
2110 GOTO 1870                           : REM OTHERWISE TREAT AS OK

2120 REM IS IT RELATIVE?

2130 IF LEFT$(C$,1) <> "B" GOTO 2400
2140 S$ = C$
2150 GOSUB 3820                          : REM FIND OPCODE
2160 IF O6<0 GOTO 1130
2170 GOSUB 3150                          : REM EVALUATE OPERAND
2180 IF O7 = -1 GOTO 1110
2190 IF O7 = -2 GOTO 2280                : REM DEFER
2200 O7 = O7 - L8 -2
2210 IF O7 >= - 128 GOTO 2240
2220 PRINT "BRANCH OUT OF RANGE"
2230 GOTO 1080
2240 IF O7 >127 GOTO 2220
2250 IF O7<0 THEN O7 = O7 + 256
2260 GOTO 1870

2270 REM DEFER RELATIVE

2280 FOR I = 1 TO D6                     : REM SEARCH FOR EMPTY SPOT
2290      IF D8(I) = -1 GOTO 2330
2300 NEXT I
2310 PRINT "DEFERRED OPERAND OVERFLOW"
2320 GOTO 1080
2330 B = O6
2340 GOSUB 3640                          : REM CONVERT 2 HEX
2350 D$(I) = "2"+B$+O$
2360 D8(I) = L8
2370 Y$ = ".."
2380 GOTO 1900

2390 REM SEPARATE OUT INHERENT FROM EXTENDED OR DIRECT

2400 S$ = C$
2410 GOSUB 3820                          : REM FIND OPCODE
2420 IF O6<0 GOTO 2500
2430 B = O6
2440 GOSUB 3640                          : REM CONVERT 2 HEX
2450 X$ = B$
2460 IF L1 = 1 THEN GOSUB 2950           : REM PRINT ML AND AL
2470 L8 = L8 + 1
2480 GOTO 270

2490 REM DIRECT INSTRUCTION?

2500 GOSUB 3150                          : REM EVALUATE OPERAND
```

SIC, since most BASICs simply assume that you should stop when you get to the last line of a program. The computer does not do that while executing a machine language program.)

Another way to add the numbers 5 and 2 would be to place them somewhere into memory and then refer to them by their addresses. For example, this program would do the job:

1000	B6	100C	Load the contents of 100C into the accumulator
1003	DD	100B	Add the contents of 100B
1006	B7	2000	Store the result in location 2000
1009	7E	1009	GOTO 1009 to stop
100C	05		First number
100D	02		Second number

Notice how the numbers to be used in this case immediately follow the program itself; there is no reason why they cannot be placed here—as long as the numbers do not appear in the midst of the program. Note that the program will never get past the loop in 1009, so placing data starting at 100C is safe.

In the previous example, I first cleared the accumulator with the 4F (CLR A) instruction, and then added both numbers to it. This time I used a load instruction, which automatically clears the accumulator and then puts the first number into it. This saves an extra instruction and makes the program faster.

A Bit of Homework

So far, I have been discussing machine language programming. There is no doubt that programming in machine language is not easy. Fortunately, you do not have to do it very often. Any reasonably complete computer system will have an assembler program available which allows you to program in assembly language. An assembler takes much of the drudgery out of machine programming by doing some of the more difficult jobs itself.

I will continue with assembly language programming next month. In the meantime, if you have access to a computer that runs BASIC and has strings, enter Program 1 into the machine in preparation for the next installment. (If you do not feel like typing it in, cassettes in either Kansas City format or in TRS-80 Level II format are available for $9.95 from Star-Kits, PO Box 209, Mt. Kisco, NY 10549.)

This program is a 6802 cross-assembler. The term cross means that this assembler runs on a computer different from the one it translates programs for. In this case, this assembler will translate 6802 assembly language into machine language, but since it is itself written in BASIC, it can be run on virtually any other machine.

We will use this program next time to assemble some simple 6802 programs for our Kilobaud Klassroom Komputer. See you then.■

```
2510 IF O7<0 GOTO 2580                    : REM DO EXTENDED IF DEFER
2520 IF O7>255 GOTO 2580                  : REM CAN'T BE DIRECT
2530 S$ = C$ + "D"
2540 GOSUB 3820                           : REM FIND OPCODE
2550 IF O6<0 GOTO 2580                    : REM DO EXTENDED IF NO OPCODE
2560 GOTO 1870                            : REM OTHERWISE TREAT SAME
                                            AS INDEXED
2570 REM EXTENDED OR 2-BYTE IMMEDIATE

2580 S$ = C$ + "E"
2590 GOSUB 3820                           : REM FIND OPCODE
2600 IF O6<0 GOTO 1130
2610 IF O7 = -1 GOTO 1080
2620 IF O7 = -2 GOTO 2740                 : REM GO DEFER
2630 A = O7
2640 GOSUB 3730                           : REM CONVERT 4 HEX
2650 Y$ = A$
2660 Z$ = ""
2670 B = O6
2680 GOSUB 3640                           : REM CONVERT 2 HEX
2690 X$ = B$
2700 IF L1 = 1 THEN GOSUB 2950            : REM PRINT ML AND AL
2710 L8 = L8 + 3
2720 GOTO 270

2730 REM DEFER EXTENDED OR TWO-BYTE IMMEDIATE

2740 FOR I = 1 TO D6                      : REM SEARCH FOR EMPTY
2750      IF D8( I ) = -1 GOTO 2780
2760 NEXT I
2770 GOTO 2310                            : REM ALL FULL
2780 B = O6
2790 GOSUB 3640                           : REM CONVERT 2 HEX
2800 D$( I ) = "3" + B$ + O$
2810 D8( I ) = L8
2820 Y$ = ".."
2830 Z$ = ".."
2840 GOTO 2670

2850 REM SUBROUTINE TO PRINT ML CODE LESS CR

2860 A=L8
2870 GOSUB 3730                           : REM CONVERT LOC TO HEX
2880 PRINT #P4, A$; "  "; X$; " "; Y$; Z$;
2890 RETURN

2900 REM SUBROUTINE TO PRINT ML CODE ONLY
2910 GOSUB 2860
2920 PRINT #P4
2930 RETURN

2940 REM SUBROUTINE TO PRINT ML AND AL CODE

2950 GOSUB 2860                           : REM PRINT ML AND THEN CONTINUE

2960 REM SUBROUTINE TO PRINT AL CODE ONLY

2970 PRINT #P4, TAB( 16); L$; TAB( 23); C$; TAB( 29 );T$+O$; TAB( 36 ); R$
2980 RETURN

2990 REM SUBROUTINE TO FIND LABEL

3000 A8 = -2
3010 IF L7 = 0 THEN RETURN
3020 FOR L5 = 1 TO L7
3030      IF A$ = N$( L5 ) GO TO 3060
3040 NEXT L5
3050 RETURN
3060 A8 = L9( L5 )
3070 RETURN

3080 REM SUBROUTINE TO SAVE LABEL

3090 IF L$ = "" THEN RETURN
3100 L7 = L7 + 1
3110 N$( L7 ) = L$
3120 L9( L7 ) = L8
3130 RETURN

3140 REM SUBROUTINE TO EVALUATE OPERAND

3150 O7 = -1
3160 IF O$ = "" THEN RETURN

3170 REM CHECK FOR HEX OPERAND

3180 IF LEFT$(O$,1) <> "$" GOTO 3280
3190 O7 = 0
3200 FOR I = 2 TO 5
3210      A$ = MID$( O$, I, 1)            : REM NEXT DIGIT
3220      GOSUB 3560                      : REM CONVERT HEX DIGIT
3230      IF A<0 THEN RETURN
3240      O7 = O7*16 + A
```

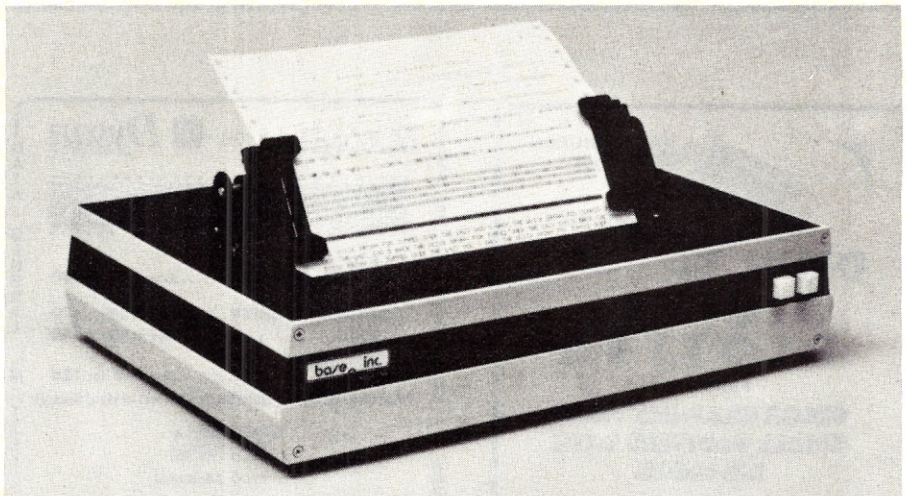
```
3250 NEXT I
3260 RETURN

3270 REM CHECK FOR AN ASCII CHARACTER

3280 IF LEFT$(O$,1) <> "'" GOTO 3320
3290 O7 = ASC(MID$(O$,2,1))
3300 RETURN

3310 REM CHECK FOR DECIMAL NUMBER

3320 A$ = LEFT$(O$, 1)
3330 GOSUB 3500                              : REM CONVERT DECIMAL
3340 IF A<0 GOTO 3440
3350 O7 = A
3360 FOR I = 2 TO 5
3370      A$ = MID$(O$,I,1)
3380      GOSUB 3500                         : REM CONVERT DECIMAL
3390      IF A<0 THEN RETURN
3400      O7 = O7*10 + A
3410 NEXT I
3420 RETURN

3430 REM FINALLY, LOOK FOR LABEL

3440 O7 = -2
3450 A$ = O$
3460 GOSUB 3000                              : REM FIND LABEL
3470 O7 = A8
3480 RETURN

3490 REM SUBROUTINE TO CONVERT DECIMAL DIGIT

3500 A = -1
3510 IF A$ < "0" THEN RETURN
3520 IF A$ > "9" THEN RETURN
3530 A = VAL(A$)
3540 RETURN

3550 REM SUBROUTINE TO CONVERT HEXADECIMAL DIGIT

3560 GOSUB 3500                              : REM CONVERT DECIMAL
3570 IF A>= 0 THEN RETURN
3575 IF A$="" GOTO 3610
3580 A = ASC(A$) - ASC ("A") + 10
3590 IF A<10 GOTO 3610
3600 IF A<16 THEN RETURN
3610 A = -1
3620 RETURN

3630 REM SUBROUTINE TO CONVERT TO 2 HEX DIGITS

3640 B = B - INT(B/256) * 256                : REM MODULO 256
3650 C = INT(B/16)
3660 B = B - C*16 + 48
3670 C = C + 48
3680 IF B>57 THEN B = B+7
3690 IF C>57 THEN C = C+7
3700 B$ = CHR$(C) + CHR$(B)
3710 RETURN

3720 REM SUBROUTINE TO CONVERT TO 4 HEX DIGITS

3730 B = INT(A/256)                          : REM LEFT TWO
3740 A = A - B*256                           : REM RIGHT TWO
3750 GOSUB 3640                              : REM CONVERT LEFT TWO
3760 A$ = B$
3770 B = A
3780 GOSUB 3640                              : REM CONVERT RIGHT TWO
3790 A$ = A$ + B$
3800 RETURN

3810 REM SUBROUTINE TO FIND OP-CODE

3820 O6 = -1
3830 FOR I = 1 TO 256
3840      READ A$
3850      IF A$ = S$ GOTO 3890               : REM FOUND IT
3860 NEXT I
3870 RESTORE
3880 RETURN
3890 O6 = I-1
3900 RESTORE
3910 RETURN

3920 REM INSTRUCTION CODE TABLE

3930 DATA -,NOP,-,-,-,-,TAP,TPA
3940 DATA INX,DEX,CLV,SEV,CLC,SEC,CLI,SEI
3950 DATA SBA,CBA,-,-,-,-,TAB,TBA
3960 DATA -,DAA,-,ABA,-,-,-,-
3970 DATA BRA,-,BHI,BLS,BCC,BCS,BNE,BEQ
3980 DATA BVC,BVS,BPL,BMI,BGE,BLT,BGT,BLE
```

```
3990 DATA TSX,INS,PULA,PULB,DES,TXS,PSHA,PSHB
4000 DATA -,RTS,-,RTI,-,-,WAI,SWI
4010 DATA NEGA,-,-,COMA,LSRA,-,RORA,ASRA
4020 DATA ASLA,ROLA,DECA,-,INCA,TSTA,-,CLRA
4030 DATA NEGB,-,-,COMB,LSRB,-,RORB,ASRB
4040 DATA ASLB,ROLB,DECB,-,INCB,TSTB,-,CLRB
4050 DATA NEGX,-,-,COMX,LSRX,-,RORX,ASRX
4060 DATA ASLX,ROLX,DECX,-,INCX,TSTX,JMPX,CLRX
4070 DATA NEGE,-,-,COME,LSRE,-,RORE,ASRE
4080 DATA ASLE,ROLE,DECE,-,INCE,TSTE,JMPE,CLRE
4090 DATA SUBA#,CMPA#,SBCA#,-,ANDA#,BITA#,LDAA#,-
4100 DATA EORA#,ADCA#,ORAA#,ADDA#,CPX#,BSR,LDS#,-
4110 DATA SUBAD,CMPAD,SBCAD,-,ANDAD,BITAD,LDAAD,STAAD
4120 DATA EORAD,ADCAD,ORAAD,ADDAD,CPXD,-,LDSD,STSD
4130 DATA SUBAX,CMPAX,SBCAX,-,ANDAX,BITAX,LDAAX,STAAX
4140 DATA EORAX,ADCAX,ORAAX,ADDAX,CPXX,JSRX,LDSX,STSX
4150 DATA SUBAE,CMPAE,SBCAE,-,ANDAE,BITAE,LDAAE,STAAE
4160 DATA EORAE,ADCAE,ORAAE,ADDAE,CPXE,JSRE,LDSE,STSE
4170 DATA SUBB#,CMPB#,SBCB#,-,ANDB#,BITB#,LDAB#,-
4180 DATA EORB#,ADCB#,ORAB#,ADDB#,-,-,LDX#,-
4190 DATA SUBBD,CMPBD,SBCBD,-,ANDBD,BITBD,LDABD,STABD
4200 DATA EORBD,ADCBD,ORABD,ADDBD,-,-,LDXD,STXD
4210 DATA SUBBX,CMPBX,SBCBX,-,ANDBX,BITBX,LDABX,STABX
4220 DATA EORBX,ADCBX,ORABX,ADDBX,-,-,LDXX,STXX
4230 DATA SUBBE,CMPBE,SBCBE,-,ANDBE,BITBE,LDABE,STABE
4240 DATA EORBE,ADCBE,ORABE,ADDBE,-,-,LDXE,STXE
```

A Printer with Panache

The Model 800B from Base 2, Inc., combines an impressive array of features with low cost.

R. A. Geanangel
11415 Kirkmeadow
Houston, TX 77089



The Base 2 Model 800B printer.

One of the most frustrating aspects of the microcomputer revolution is that although you can buy a surprisingly powerful computer for around $500, a hard-copy unit for it costs nearly twice as much. Recently, however, things have changed for the better. Advanced line printers with many desirable features are now available inexpensively. One combines both a low price ($699) and an impressive array of features—the Model 800B from Base 2, Inc.

General Features

The Base 2 printer is a dot matrix impact printer that produces a 96-character (upper and lowercase) ASCII set. Line lengths of 72, 80, 96, 120 and 132 characters may be selected by switch-setting or through software. The printer has capabilities for RS-232 and 20 mA current loop serial interfaces along with Centronics and IEEE-488 parallel interfaces, all available through connectors on the rear panel and all switch-selectable. Optional cables are available from the manufacturer.

Sixteen baud rates, with a maximum of 19,200, can be selected, using a convenient rotary switch, also on the rear panel. The printer also has a self-test mode, which can be operated independently of any external connection.

Mechanically, the Model 800B is very simple. Paper up to 9½ inches wide is fed into the printer from the bottom of the unit and moves upward with tractor or friction feed. Paper advance is activated by one of two buttons on the front panel. (The other button controls unit select.) The printing element is moved across the paper by a spiral-grooved cylinder. Printing is bidirectional, and the print head has a life expectancy in excess of 100 million characters.

Three features which previously were options on the Model 800 are:

- A 1920-character terminal buffer.
- A stepper motor for high-speed paper advance and dot resolution graphics.
- A tractor feed mechanism which, in concert with the previous option, can be used to create hard-copy graphics output.

Print Format Features

The Base 2 printer owes its flexibility in large part to the use of an 8085 microprocessor and 32K ROM of control firmware. The printer responds to a selection of the usual ASCII control codes, permitting printing format versatility. In addition, there are a variety of special function codes, each of which must be preceded by an ASCII ESC (1B hex) code. The printer responds directly to the ordinary control codes, including carriage return, line feed, form feed and vertical tab.

You can also select and deselect the printer under software control and cause special horizontally elongated characters to be printed—even intermixed—on a single line with regular characters.

Features such as these are fairly common among competing printers, but the surprises are found in the other software-controlled features of the Model 800B.

Table 1 lists the special format functions supported by the printer. First, the horizontal print density can be changed in steps from eight to 16½ characters per inch (64 to 132 characters per line). Perhaps the most remarkable feature of the printer is its accommodation of multiple character fonts. Besides the normal and elongated, upper and lowercase, the standard version of the printer sports the APL set in ROM, invoked by the appropriate function code.

But that's not all. You may define your own character set, and down-load it from the computer. The format involves a 5×7 dot matrix format and is straightforward, if somewhat tedious, to implement. If that weren't enough, you can define up to eight additional character sets, using an EPROM, for which space is provided on the printer's

ASCII Code	Decimal Code	Function
ESC	27	Advises printer of new command sequence
0	48	Sets line length to 72 characters (9 cpi)
1	49	Sets line length to 80 characters (10 cpi)
2	50	Sets line length to 96 characters (12 cpi)
3	51	Sets line length to 120 characters (15 cpi)
4	52	Sets line length to 132 characters (16.5 cpi)
@	64	Enables elongated character mode
A	65	Disables elongated character mode
B	66	Enables recognition of CR by printer
C	67	Disables recognition of CR by printer
D	68	Enables recognition of LF by printer
E	69	Disables recognition of LF by printer
F,n	70,n	Sets paper to be ejected "n" lines
G	71	Causes paper to be ejected
H	72	Enables the printer to receive data
I	73	Sets the printer off-line
J	74	General reset to initialization parameters
K	75	Loads character set in auxiliary font buffer
L	76	Enables user defined character font
M	77	Enables standard character font
N	78	Enables secondary character font
O	79	Enables optional character font 1
P	80	Enables optional character font 2
Q	81	Enables optional character font 3
R,n,m	82,n,m	Sets buffer length to value loaded
S	83	Prints buffer contents
T,n	84,n	Set lines per page and lines to skip to new page
V,$n_1 \ldots n_{10}$	86,$n_1 \ldots n_{10}$	Sets horizontal tab positions—up to ten
X	88	Resets all tabs
Y,$n_1 \ldots n_{10}$	89,$n_1 \ldots n_{10}$	Sets up to ten vertical tab positions
a	97	Resets all tabs
b,n	98,n	Sets vertical line spacing to n dots
c,data	99,data	Transmits graphics data
5	53	Disable print on Buffer Full
6	54	Enable print on Buffer Full
7	55	Enable Auto LF with CR
8	56	Disable Auto LF with CR
9	57	Set Auto FF count
:	58	Enables Auto FF
;	59	Disable Auto FF

Table 1. Function codes for special features.

logic board.

Graphics Feature

The possibility of graphics printout under software control is another unusual feature of the Base 2 printer. Operation in this mode is accomplished by the use of the stepper motor to give precision control of paper advance.

In effect, graphics printing is carried out by eliminating the vertical spacing between lines and the horizontal spacing between characters and simply outputting a stream of characters. The manufacturer warns, however, that extensive graphics output can overheat and possibly damage the printhead.

Using the Model 800B

My experiences with the Model 800B printer began with a nine-week wait for it to be delivered. The Base 2 folks missed the estimated shipping date by only ten days, which is not bad considering the shortage of components we hear about. My budget dictated that I order the standard model without any extras.

The unit has an air of solidity about it. The moving parts are few, and the logic board is well-organized with a minimum of point-to-point wiring.

Upon receipt of the printer, I set about connecting it to my TRS-80 computer, a 32K Model I with a single disk. Since the expansion interface provided a convenient parallel port for a line printer, I choose to employ it for the I/O connection. Fortunately, my spare parts drawer contained the necessary connectors. The wiring interconnections are listed in Table 2. Note that the Base 2 printer doesn't provide an out-of-paper signal; that line on the TRS-80 expansion interface parallel port must be tied to ground. After connecting that oversight, I was able to LPRINT and LLIST immediately.

The printing is fast (about one line per second according to the manual) and readable, except, perhaps, at 132 characters per line (see Sample Run). The printer is moderately noisy, but no more than other comparable units I have worked with, such as the Heathkit printer. I used about two feet of flat ribbon cable in my I/O connection, with ground lines separating data lines, and there was no evidence of noise problems in the printout.

My intentions are to use the Base 2 printer in scientific programs that use statistics, curve-fitting and various types of calculations. I therefore tested its special print formatting features.

I did this initially by taking advantage of the fact that the line printer I/O on my computer is memory mapped. Thus, it was easy to write a short BASIC program to poke control codes and function codes into the appropriate memory location to investigate the printer's response. The Sample Run illustrates the character format control the Model 800B provides.

The first line illustrates the elongated character mode obtained by transmitting decimal 14 (CTL N) to the printer. The following six lines illustrate the character set in 64, 72, 80, 96, 120 and 132 characters per line.

An interesting feature to those with word-processing and related applications is the auto form feed capability. The printer may be programmed to skip six lines after a predetermined number of lines has been printed. You need only to send the auto form-feed line count to the printer, following it with the appropriate function code to enable the auto form-feed mode. You can also skip a predetermined number of lines without issuing individual line feeds. Tractor feed, as opposed to friction feed, should be used if you are to take proper advantage of these features.

Base 2 Printer Parallel Connector (Pin No.)	TRS-80 Expansion Interface Line Printer-Connector (Pin No.)	Function
1	1	Data strobe
2	3	Data bit 1
3	5	Data bit 2
4	7	Data bit 3
5	9	Data bit 4
6	11	Data bit 5
7	13	Data bit 6
8	15	Data bit 7
14,16,17,19–30, 33	2,4,6,8,10,12,14,16,18,20,22,24	Ground
11	21	Busy
13	25	Unit select

Note: Pin 23 on the expansion interface represents an out-of-paper condition input from the printer. This is grounded, since no such output is available from the printer.

Table 2. Wiring connections between TRS-80 and Base 2 printer.

The Base 2 organization has recently released a first-rate manual for their Model 800B printer. Five sections covering general information, functional description, installation and operation, software control and interface specifications occupy 44 pages. Included are photographs of all mechanical parts in exploded views. TRS-80 BASIC routines are given for the implementation of the printer's various features. Appendices including full schematics, parts lists and timing diagrams comprise the last 24 pages of this excellent manual.

All things considered, the Base 2 printer has an impressive array of capability for the price. I look forward to using it in my present and future programs.■

Base 2, Inc.
PO Box 3548
Fullerton, CA 92634

```
          BASE  2  PRINTER  TEST

  64 CHAR. PER LINE. !"##$%&'()*+,-./0123456789:;<=>?@AB
LMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~▒

  72 CHAR. PER LINE. !"##$%&'()*+,-./0123456789:;<=>?@ABCDEFGHI
TUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~▒

  80 CHAR. PER LINE. !"##$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOP
\]^_`abcdefghijklmnopqrstuvwxyz{|}~▒

 96 CHAR. PER LINE. !"##$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
mnopqrstuvwxyz{|}~▒

120 CHAR. PER LINE. !"##$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrst

132 CHAR. PER LINE. !"##$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~▒
```

Sample Run.

ITEM NO.
WK-7

CMOS SAFE

IC INSERTION/EXTRACTION KIT

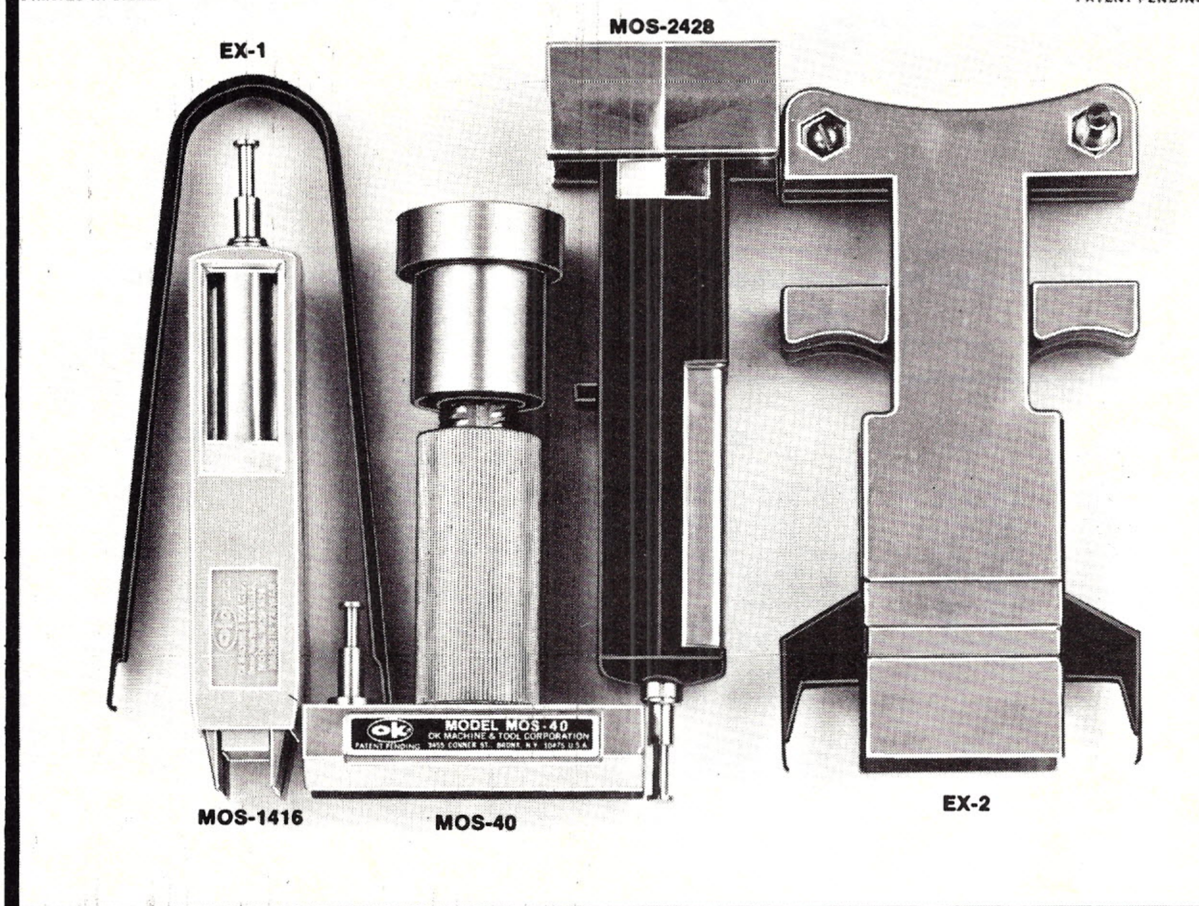
KIT INCLUDES

- **MOS-1416** 14-16 CMOS SAFE INSERTER
- **MOS-2428** 24-28 CMOS SAFE INSERTER
- **MOS-40** 36-40 CMOS SAFE INSERTER
- **EX-1** 14-16 EXTRACTOR
- **EX-2** 24-40 CMOS SAFE EXTRACTOR

✓27

OK MACHINE & TOOL CORPORATION
3455 CONNER ST., BRONX, N.Y. 10475 U.S.A.
PHONE (212) 994-6600 TELEX NO 125091

PRINTED IN U.S.A. PATENT PENDING



WK-7	COMPLETE IC INSERTER/EXTRACTOR KIT	$29.95

INDIVIDUAL COMPONENTS

MOS-1416	14-16 PIN MOS CMOS SAFE INSERTER	$ 7.95
MOS-2428	24-28 PIN MOS CMOS SAFE INSERTER	$ 7.95
MOS-40	36-40 PIN MOS CMOS SAFE INSERTER	$ 7.95
EX-1	14-16 PIN EXTRACTOR TOOL	$ 1.49
EX-2	24-40 PIN CMOS SAFE EXTRACTOR TOOL	$ 7.95

MINIMUM BILLING $25.00. ADD SHIPPING CHARGE $2.00. NEW YORK RESIDENTS ADD APPLICABLE TAX.

OK MACHINE & TOOL CORPORATION 3455 CONNER ST., BRONX, N.Y. 10475 (212) 994-6600/TELEX 125091

Tinkering with Tiny BASIC

How to add four new and useful commands to Tom Pittman's brainchild, plus some tips on using USR.

Michael L. Bugg
396 Birdcage Walk
Mansfield, OH 44903

DECIMAL	HEX	USE
0032-0033	0020-0021	START OF BASIC PROGRAM (POINTER)
0034-0035	0022-0023	END OF USER MEMORY (POINTER)
0036-0037	0024-0025	END OF BASIC PROGRAM (POINTER)
0038-0039	0026-0027	TOP OF BASIC STACK (POINTER)
0040-0041	0028-0029	CURRENT LINE NUMBER
0042-0043	002A-002B	I. L. PROGRAM COUNTER
0044-0045	002C-002D	BASIC POINTER
0046-0047	002E-002F	SAVED POINTER
0048-0127	0030-007F	LINE INPUT & EXPRESSION STACK
0128-0129	0080-0081	RANDOM NUMBER SEED
0130-0181	0082-00B5	VARIABLES: 2 bytes each in order A @ 0130-0131 B @ 0132-0133 : Z @ 0180-0181
0191	00BF	OPT COLUMN COUNTER & TAPE MODE
0256	0100	TEST FOR BREAK ROUTINE
0512	0200	COLD START-TINY BASIC
0515	0203	WARM START-TINY BASIC
0518-0520	0206-0208	JMP TO GET CHARACTER
0521-0523	0209-020B	JMP TO PRINT CHARACTER
0524-0526	020C-020E	JMP TO BREAK TEST
0532	0214	READ MEMORY BYTE SUB (PEEK)
0536	0218	STORE MEMORY BYTE SUB (POKE)
2416	0970	START OF IL CODE
2816	0B00	BASIC PROGRAM STARTS HERE-NORMAL
2897	0B51	START OF SCRATCH-PAD AREA IN MY MODIFIED TINY BASIC
3072	0C00	BASIC PROGRAM STARTS HERE-MODIFIED
6016-6118	1780-17E6	KIM: EXTRA USEABLE MEMORY
7168	1C00	KIM: START OF KIM MONITOR
7739	1E3B	KIM: PRINT HEX BYTE
7838	1E9E	KIM: PRINT SPACE
7840	1EA0	KIM: PRINT ASCII CHARACTER
7770	1E5A	KIM: INPUT ASCII CHARACTER
8093	1F9D	KIM: INPUT HEX BYTE

Table 1. Tiny BASIC decimal reference chart.

```
5    REM  ENTER HEX BYTE, PRINT DECIMAL EQUIVALENT
10   PRINT "ENTER HEX BYTE "
20   LET X = USR (8093)
30   PRINT" ";
40   PRINT X
50   END
```

Listing 1.

This article describes the USR function of Tiny BASIC and shows you how to add a few new commands to facilitate writing programs so you can replace the USR function in many instances with more understandable coding. I have also included some information and hints I found useful in tinkering with Tiny BASIC (both in modifying it and using it).

I bought a KIM-1 several years ago, but, being an avid do-it-yourselfer, I never thought I would ever buy software. I became tired of keying in programs and accidentally wiping them out by miscalculating a relative branch or missing a byte.

Tom Pittman's Tiny BASIC solved these problems. For those of us with small systems, it has to be the best software buy around. It fits quite comfortably in my 4K

```
5    REM  ENTER HEX BYTE, PRINT DECIMAL EQUIVALENT
10   PRINT "ENTER HEX BYTE ";
20   PRINT USR (8093)-0*USR(7838)
30   END
```

Listing 2.

```
5    REM  ENTER 2 HEX BYTES, PRINT DECIMAL EQUIVALENT
10   PRINT "ENTER 2 HEX BYTES ";
20   PRINT 256*USR(8093)+USR(8093)-0*USR(7838)
30   END
```

Listing 3.

memory, with room enough for my limited collection of games. (I plan on expanding the memory sometime, but for now, Tiny BASIC is it.)

Using USR

One feature of Tiny BASIC that provides unlimited versatility is the USR function. However, it was some time before I actually realized its potential. At first, I was hesitant to use it, due in part to having to calculate the addresses (and any other normally hex numbers) into decimal. However, using KIM's built-in subroutines, you can program KIM-1 (in Tiny BASIC) to perform the hex to decimal conversion for you.

The USR function is simply a machine-language subroutine call. A language such as Tiny BASIC is capable of performing almost anything you want it to do, but in some instances a machine-language subroutine is more expeditious. So, Tiny's USR is the way to break out of BASIC and execute a machine-language subroutine directly.

Listing 1 shows a simple Tiny BASIC program written for KIM using the USR function. (Other systems require adjusting the address, which this program jumps to.) This program uses one of KIM's built-in ROM monitor subroutines: the input hex byte routine (GETBYT in the KIM-1 monitor assembly listing). With this subroutine, Listing 1 converts a hex byte into its decimal equivalent.

CHARACTER	DECIMAL	HEX	DECIMAL X 2
A	65	41	130
B	66	42	132
C	67	43	134
D	68	44	136
E	69	45	138
F	70	46	140
G	71	47	142
H	72	48	144
I	73	49	146
J	74	4A	148
K	75	4B	150
L	76	4C	152
M	77	4D	154
N	78	4E	156
O	79	4F	158
P	80	50	160
Q	81	51	162
R	82	52	164
S	83	53	166
T	84	54	168
U	85	55	170
V	86	56	172
W	87	57	174
X	88	58	176
Y	89	59	178
Z	90	5A	180
0	48	30	
1	49	31	
2	50	32	
3	51	33	
4	52	34	
5	53	35	
6	54	36	
7	55	37	
8	56	38	
9	57	39	
+	43	2B	
-	45	2D	
/	47	2F	
*	42	2A	
.	46	2E	
RETURN	13	0D	

Table 2. Decimal equivalents.

Line 10 prints the instruction to the operator. In line 20, the variable X is made equal to whatever value is in the system accumulator upon return from the subroutine addressed by the following USR function. In this case, the value is the hex byte value obtained by packing two hex digits entered on the terminal keyboard. (Typing on the keyboard produces an ASCII-code byte for each digit entered, so this routine converts and packs them into one byte for each two entered, with the resultant byte in the accumulator register.)

When Tiny gets to this USR, it will jump to decimal address 8093, which is 1F9D in hex, the start of the GETBYT subroutine (remember: Tiny uses decimal numbers, so you will need to know the decimal equivalent of the address being jumped to). When Tiny gets here, the computer waits for the operator to punch in two hex digits on the keyboard. After the second key is accepted, the data is packed and returned to Tiny, where the variable X becomes this hex value.

Line 30 simply prints a space to separate the hex entry from the computer's upcoming response. A semicolon at the end of the line keeps everything on the same line. Line 40 prints the value held in variable X. Although we entered a hex value, Tiny BASIC prints its decimal equivalent. Thus, we have a program to convert from hex into decimal.

Listing 2 does exactly the same thing as Listing 1. Line 10 prints the instruction to the operator. In line 20 the PRINT command tells Tiny to print the value of the expression that follows. First, it evaluates the expression. USR (8093) comes first, so we jump to this address (just as before) to get the hex input.

The subroutine returns control back to Tiny, and the program continues. So far, the expression's value is the hex number we entered on the keyboard. The second half of the expression in line 20 starts out by subtracting zero times the value of USR (7838), which is the same as subtracting nothing. This assures that our previous value obtained will be left unchanged.

Now Tiny jumps to the subroutine at decimal 7838 (hex 1E9E). This is the system monitor's print-a-space subroutine (OUTSP in the KIM monitor listing). Keep in mind that the hex byte was already printed when we entered it through the terminal. When this second USR is executed, a space is placed just after the hex byte. Following this, we again return to Tiny, and, being at the end of the expression, the resultant value is finally printed. Since we zeroed out the second USR (assuming that the data returned in the accumulator will be useless and unknown), it has no effect on the expression's value, and our original hex number remains to be converted into decimal and printed.

This program shows what you can do with the USR function to save a little

memory space. By combining operations onto fewer program lines in this fashion, we can save that precious space in super small systems, where every byte counts.

Computing Two-Byte Addresses

Most addresses in the computer take two bytes to define, so we need to make the expression equal to a value of four hex digits entered. By modifying line 20 of Listing 2, we can create a program to convert out known hex addresses into decimal, expedite writing out those USR functions and have Tiny BASIC do our work for us.

The modification is shown in Listing 3. Note that because the subroutine called by USR (8093) only accepts one byte at a time, we must call it twice to get what we need. The first call obtains the most significant byte (MSB), so we multiply it by 256, which effectively shifts it into the proper position so Tiny evaluates it the way we want. The next call produces the least significant byte (LSB), which we add to what we already have. Finally, a call is made (as in the previous program) to print the space. The value is printed in decimal.

Using this decimal address calculator (as well as any other program using such subroutines), you must enter all four hex digits (or two for the earlier programs), including any leading zero. Also, because it is a machine-language subroutine (outside of Tiny BASIC), no input prompt is offered, and you don't have to hit the return key after entering the input. You may, of course, in a PRINT statement preceding such an input, cause a prompt of any sort to be printed.

I have used this program to work up a chart of often-used decimal addresses (Table 1). Also, a list of decimal values for some of the commonly used ASCII characters is convenient for testing data in the input buffer (Table 2). These tables, as well as this article, deal mainly with Tiny BASIC as run on a KIM-1 system starting at hex address 0200. For other addresses at which Tiny may be loaded, or other systems not having the monitor routines as listed, you would have to modify the program (but with the decimal address conversion program, this should be no problem).

Table 2 contains a column with decimal values times two. Tiny stores its variables in an address equivalent to the ASCII value of the variable name (alpha-character) doubled. For example, the location of variable A in hex is 82 (the ASCII value of A is 41, which doubled is 82) or 130 (65 times two) in decimal.

Machine-Language Programming

Tiny BASIC's ability to stay together even if I make a programming mistake, along with KIM's built-in monitor subroutines, proves to be a great aid in machine-language programming. You can first program and debug complicated algorithms in BASIC and then translate them into machine language. It's easier to delete an instruction or modify the program in BASIC

```
5     REM  HEX RELATIVE BRANCH CALCULATOR
6     REM  I= INPUT HEX SUB     S= PRINT SPACE SUB
10    I=8093
20    S=7838
30    PRINT "ENTER 'TO' THEN 'FROM' ADDRESSES (2HEX BYTES EACH) "
40    Z= USR(7739,0,256*USR(I)+USR(I)+USR(S)-256*USR(I)-USR(I)-
       USR(I)-USR(S)-2)
45    REM  USR(7739  PRINTS HEX BYTE
50    END
```

Listing 4.

than to rewrite a machine-language program to make a few changes. Once the program works properly, you can put it into machine language with Tiny helping out. Tiny BASIC can do your relative branch calculations for you. Listing 4 shows how.

Listing 4 accepts two four-digit hex addresses, automatically separates them with a space, and then prints the relative branch operand in hex. To conserve space, I used variables for the input (I) and print space (S) subroutine addresses. Table 3 summarizes the features of the USR function.

The USR functions are commonly used for two subroutines built into Tiny for reading and storing a memory byte, as PEEK and POKE in other BASICs. Although these are useful, they have one drawback: they can be difficult to follow if there are multiple USRs nested within USRs. If I review a program I had written some time ago, it takes me awhile to figure out what I had done. So, I decided to make Tiny a little bit bigger.

Adding @ and &

To make writing programs more understandable when imitating the PEEK and POKE functions of other BASICs, I modified Tiny to include a couple of new operators—@ (for one-byte numbers) and & (for two-byte numbers). Adding these to Tiny is easy.

Consider the following program line using standard Tiny BASIC syntax:

P = P − 0 * USR (538, USR (534, 46), 13)

This stores a carriage return (decimal 13) in the memory location pointed to by the line pointer (decimal 46). This is used to input string data by fooling Tiny into thinking it has come to the end of the line so that the next time an INPUT command occurs a prompt will be issued and the next input will be accepted.

Consider the following line:

LET@ @ 46 = 13

This does exactly the same thing as the previous line with the USR operation in it. This line affects no variables (normally, a USR will when used as above, so we used the "multiply by zero" trick to avoid it, such as might be necessary in a program where all variables are dedicated to something else), takes up far less program memory space and is simpler to understand at a glance.

This line uses two separate operations: the LET@ and the @ functions. These are referred to as indirection operators (from Tom Pittman's Tiny BASIC Experimenter's

FORMAT:	USR (expression)
or:	USR (expression, expression)
or:	USR (expression, expression, expression)
USE:	machine-language subroutine call, jumps to the address defined by the first expression (in *decimal*)
2nd EXPRESSION:	if included, is deposited into the processor's index registers —most significant byte goes into X-index —least significant byte goes into Y-index —(remember, all expressions become two-byte values)
3rd EXPRESSION:	if included, is deposited into the processor's accumulator register (8 bits only) —most significant byte is lost —least significant byte goes into accumulator
EVALUATION	upon return to Tiny BASIC from the machine-language subroutine the USR function becomes a two-byte value which is dependent upon the following: —Y-register value becomes most significant byte —accumulator becomes least significant byte This may be expressed as: value of USR = 256 * (X-reg) + Accum
SUMMARY:	USR (address, X and Y index registers, accumulator)
USING TINY BASIC'S BUILT-IN SUBROUTINES:	—READ BYTE (PEEK): USR(532, Address) —STORE BYTE (POKE): USR(536, Address, Data)

Table 3. USR function summary.

```
PRINT @ D
LET@ 1000 = A
LET@ A = X
LET@ A = @ X + Z
IF @ E + 40 = @ X THEN GOTO @ J
LET@ @ @ X = A * @A / @ @ 46 - USR (@C, USR (@@D, 9), @2)
```

Table 4. Using @ and LET@ operations.

Kit), for a poke (store) and a peek (read), respectively. This line causes the byte at the address stored at decimal 46 to equal 13. This is a form of indirect addressing.

How Indirection Works

Suppose we want to print the value of the data at address location 1000. We must enter the command
PRINT@ 1000
This prints the data at line 1000.

You may have an indirect indirection operation:
LET X = @@46
which will cause variable X to take on whatever character the input line buffer pointer (decimal address 46) is pointing to.

To alter a specified memory byte, you must add a new keyword, LET@, to Tiny. Just as before, the number following the @ sign specifies the decimal address whose byte will be set. LET@ 1000 = 0 will set address 1000 to zero.

The @ and LET@ operations can be used in most any combination (see Table 4). Since these two operations don't exist in

0285	LSB of BASIC program starting address normally 00 I left this unchanged
028C	MSB of BASIC program starting address normally 0B I changed this to 0C
097D–097E	This becomes jump to new LET& and LET@ normally 8B-4C change to 39-90
0A91–0A92	This becomes jump to new & and @ normally C1–2F change to 39–C9

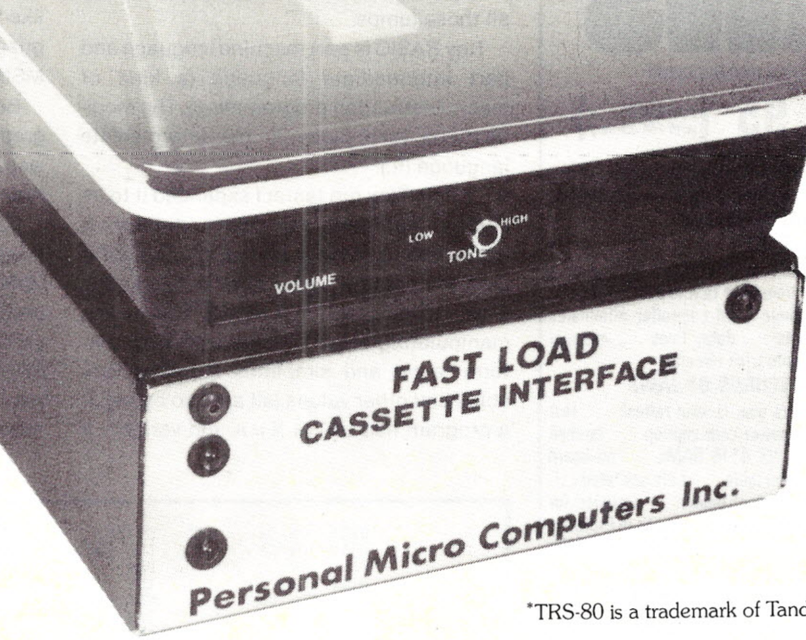
Table 5. Tiny BASIC modification changes.

```
0B00  98 4C 45 54 A6      :LET& BC LET@ "LET&"  TEST FOR LET&
0B05  0A 01 22                  LN 122          YES, SET ML ADDRESS
0B08  30 BC                     JS EXPR         GET BYTE ADDRESS
0B0A  0B                        DS
0B0B  2E                        US              GO SET ADDRESS
0B0C  0C                        SP
0B0D  0A 01 29                  LN 129          SET ML ADDRESS
0B10  80 BD                     BC * "="        TEST FOR EQUAL SIGN
0B12  30 BC                     JS EXPR         GET VALUE
0B14  0B                        DS
0B15  E0                        BE *            TEST IF LINE END
0B16  2E                        US              GO DO IT
0B17  0C                        SP
0B18  1D                        NX              END OF THIS
0B19  91 4C 45 54 C0      :LET@ BC LET "LET@"   TEST FOR LET@
0B1E  0A 02 18                  LN 218          YES, SET ML ADDRESS
0B21  30 BC                     JS EXPR         GET BYTE ADDRESS
0B23  80 BD                     BC * "="        TEST FOR EQUAL SIGN
0B25  30 BC                     JS EXPR         GET VALUE
0B27  E0                        BE *            TEST IF LINE END
0B28  2E                        US              GO DO IT
0B29  0C                        SP
0B2A  1D                        NX              END OF THIS
0B2B  8B 4C 45 D4         :LET  BC BACK "LET"   TEST FOR LET
0B2F  A0                        BV *            GET VARIABLE
0B30  80 BD                     BC * "="        TEST FOR EQUAL SIGN
0B32  30 BC                     JS EXPR         GET VALUE
0B34  E0                        BE *            TEST IF LINE END
0B35  13                        SV              PUT INTO VARIABLE
0B36  1D                        NX              END OF THIS
0B37  38 19               :BACK J GOTO          BACK TO ORIGINAL CODING
0B39  C1                  :NEW  BN F40          THIS REPLACES WHAT WAS
0B3A  2F                        RT              WIPED OUT IN ORIGINAL
0B3B  89 A6               :F40  BC F41 "&"      TEST FOR &
0B3D  0A 01 15                  LN 115          YES, SET ML ADDRESS
0B40  30 BC                     JS EXPR         GET BYTE ADDRESS
0B42  0B                        DS
0B43  2E                        US              GO DO IT
0B44  2F                        RT              RETURN
0B45  89 C0               :F41  BC RET "@"      TEST FOR @
0B47  0A 02 14                  LN 214          YES, SET ML ADDRESS
0B4A  30 BC                     JS EXPR         GET BYTE ADDRESS
0B4C  0B                        DS
0B4D  2E                        US              GO DO IT
0B4E  2F                        RT              RETURN
0B4F  39 23               :RET  J  F5           GO BACK TO ORIGINAL
                                                CODING

097D  39 90)
0A91  39 C9)--- THESE REPLACE EXISTING CODING
0B00  98 4C 45 54 A6 0A 01 22  30 BC 0B 2E 0C 0A 01 29
0B10  80 BD 30 BC 0B E0 2E 0C  1D 91 4C 45 54 C0 0A 02
0B20  18 30 BC 80 BD 30 BC E0  2E 0C 1D 8B 4C 45 D4 A0
0B30  80 BD 30 BC E0 13 1D 38  19 C1 2F 89 A6 0A 01 15
0B40  30 BC 0B 2E 2F 89 C0 0A  02 14 30 BC 0B 2E 2F 39
0B50  23
```

Listing 5.

Fastload

FOR TRS-80* MODEL I USERS ONLY



16 Times
Normal Speed

*TRS-80 is a trademark of Tandy Corp.

- High speed load TRS-80* Level II cassettes
- Input 15K byte Level II program in 15 seconds
- Search BASIC or SYSTEM programs by name

Unlike other high speed tape input devices, FASTLOAD uses standard format cassettes. Therefore, there is no need to re-record on other media. At 8000 baud, FASTLOAD is faster than disk for short programs. FASTLOAD reads tapes at the fast-forward speed of the CTR-41 cassette recorder. The recorder can also be used for CSAVE at the normal speed.

FASTLOAD connects to the 40 pin I/O or to the Expansion box. The control program does not use computer memory because it is in a built-in PROM. Other valuable features are keyboard debounce program, automatic key repeat routine and key-beep via cassette speaker. Price is $188.00 for FASTLOAD and $95.00 for the modified CTR-41 recorder.

✓112

Personal Micro Computers Inc.

475 Ellis Street, Mountain View, CA 94043 (415) 968-1604

Tiny BASIC, they must be added to it.

There are two ways of accomplishing this. The first, and best, way is to insert the coding for them into the existing interpreter at the proper points and move the following coding down with the jump addresses and adjust them accordingly. The way I do it is to blot out a part of the existing program with a jump to the new routines (which are tacked on at the end of Tiny BASIC) and have them jump back to pick up where the original coding left off. This takes a couple more bytes, but it sure beats recalculating all those jumps.

Tiny BASIC is part machine language and part intermediate language (a kind of macro-instruction programming). The modifications take place in the intermediate language (IL).

To help Tiny run faster I expanded it to include a two-byte indirection operator. It works just like the @ and LET@, except it gets and puts two bytes at a time. I use the & sign to indicate this function. This makes manipulating large amounts of data perform faster and simplifies handling variables and other values (all are two bytes). If a program had LET@ X = A and variable A was greater than 255, then part of that value would be lost (you just can't store a 16-bit value in an eight-bit byte). For timing comparisons, see Table 6.

How & and LET & Work

Suppose Z = 1. Each variable of Tiny is a two-byte value. So, in Z, the MSB is zero and the LSB is one. LET& 50 = Z will make the combined bytes 50 and 51 equal to Z. Thus, the MSB (0) will be deposited into location 51, and the LSB (1) will be put into location 50 (Tiny BASIC uses them backwards, just like the addresses in the 6502 machine-language operands: LSB comes first, then MSB).

Besides variable handling, BASIC program line numbers could be altered this way. Tables and arrays are a natural for this type of function.

To get my Tiny BASIC to learn these new things, I put the new coding at hex address 0B00 and beyond. This is where the BASIC program is normally put, so I changed the portion of Tiny that determines where the BASIC program starts. It can start just after the last byte of new coding, but I prefer to have it start at the beginning of the next

```
0115  86 C3      GET    STX$C3         STORE MSB ADDRESS  ($C2 = 0)
0117  B1 C2             LDA ($C2),Y    GET BYTE-1 (LSB ADDRESS = Y)
0119  48                PHA
011A  C8                INY
011B  B1 C2             LDA ($C2),Y    GET BYTE-2
011D  AA                TAX
011E  68                PLA
011F  A8                TAY
0120  8A                TXA
0121  60                RTS
0122  86 C3      PUT1   STX$C3         SAVE ADDRESS MSB
0124  85 E2             STA$E2                      LSB
0126  60                RTS
0127  EA EA             NOP NOP
0129  A4 E2      PUT2   LDY$E2         SET INDEX
012B  48                PHA            SAVE BYTE-2
012C  8A                TXA            LOAD BYTE-1 INTO ACCUM
012D  91 C2             STA ($C2),Y    PUT BYTE -1
012F  68                PLA
0130  C8                INY
0131  91 C2             STA ($C2),Y    PUT BYTE-2
0133  60                RTS


0115                    86 C3 B1  C2 48 CA B1 C2 AA 68 A8
0120  8A 60 86 C3 85 E2 60 EA   EA A4 E2 48 8A 91 C2 68
0130  C8 91 C2 60
```

Listing 6. Source listing for machine-language coding.

USR	@	&	USR and LET
90 M=0	M=0	M=0	LET M=0
100 N=0	N=0	N=0	LET N=0
110 P=USR(536,N,0)	LET@N=0	LET&N=0	LET P=USR(536,N,0)
120 N=N+1	N=N+1	N=N+2	LET N=N+1
130 IF N<20 GOTO 110	IF N<20 GOTO 110	IF N<20 GOTO 110	IF N<20 GOTO 110
140 M=M+1	M=M+1	M=M+1	M=M+1
150 IF M<20 GOTO 100	IF M<20 GOTO 100	IF M <20 GOTO 100	IF M<20 GOTO 100
160 PRINT "END"	PRINT "END"	PRINT "END"	PRINT "END"
170 END	END	END	END
TIME = 23 SECONDS	TIME = 18 SECONDS	TIME = 9 SECONDS	TIME = 21 SECONDS

The above four programs all perform the same duties in their own way. This serves to demonstrate how programs may be rewritten to speed things up in different ways. If a program has need to move large blocks of data (such as character strings) the LET& operation can obviously speed things up considerably.

Table 6. Timing comparison tests.

page of memory (0C00 in my system) to allow room for array storage or extra variables without interference between them and the BASIC program. (This eliminates the chances of strange things happening when a program overwrites itself.)

If you are wondering why I put the new coding starting where I did, rather than directly after the existing program (originally ending at hex 0AC6). I put a multiple-statements-per-line modification (see 6502 USER NOTES, no. 13) in this gap. After a little work, Tiny BASIC doesn't act quite so tiny!

If you want to start Tiny loading the BASIC program farther down to allow room for its new growth, you can alter 0285 and 028C (this will avoid the need to enter through the warm start and set the parameters each time you start out). Hex address 028C holds the memory page number. I set this to 0C, as opposed to 0B in the original. Address 0285 holds the LSB of the starting address (normally 0). I left this unaltered.

Listing 5 contains the new coding for all of these new operations (for the IL coding), and Listing 6 shows the additional machine coding needed to accommodate it. Finally, Table 5 shows the necessary patches to the existing coding. Again, these addresses are for BASIC starting at 0200. For other starting addresses you will need to determine the changes. After you have made this modification, refer to Table 7 to remind you how to use the new operations.

Uncluttering Your CRT

Along with printing the input prompts (: and ?) and preceding a LIST operation, Tiny BASIC outputs control codes (X-on and X-off). If your system has a CRT for readout and thus has no need for these control codes, you can replace these control codes with screen control codes to make the display more readable (without the need for extra output routines to take care of business).

My TVT doesn't scroll up as it fills the screen, so after the cursor reaches the end of the page, the following output causes the cursor to wrap around to the top again, writing over what was previously there. Sometimes, this becomes quite confusing when one line ends in the middle, leaving the remains of an old line after it. Because of this, I replaced some of Tiny's control codes with the desired screen control functions: clear

@	ONE-BYTE FETCH (PEEK)—whose value is the byte at the decimal address following the @ symbol
&	TWO-BYTE FETCH (PEEK)—whose value is the combination of the two bytes at the decimal address following the & sign (LSB) and at one plus that address (MSB)
LET@	ONE-BYTE STORE (POKE)—stores a byte at the decimal address following the @ symbol
LET&	TWO-BYTE STORE (POKE)—stores a two-byte value at the decimal address following the & sign (LSB) and at one plus that address (MSB)

The addresses specified in the above operations may be any valid expression accepted by Tiny BASIC, including other similar operations, USR functions, etc.

Table 7. @, LET@, & and LET& operations summary.

line, clear screen and cursor home.

At 0972 hex Tiny issues X-on after printing the colon prompt. Replace this with your choice of line or screen clear. (I use line clear.) If screen clear is used, when Tiny gives me an error code and the CRT is at the bottom line, the following colon and control code would be printed on the top line, thereby wiping out the error code before it can be read.

When inserting the code, you must alter it: set the highest bit to one. Thus, if your desired control code is 06, it must be set to 86 to insert Tiny.

The control code following the INPUT prompt (?) is located at 09DD hex. Again, observe the above instructions on setting the high bit to one.

At addresses 0A03–0A06 hex are four bytes that are printed preceding a LIST operation. These are normally all zeros, but I first insert a cursor-home control, followed by a clear-screen character. This way, the LIST starts automatically at the top of the screen and clears any previous clutter.

Also, within a program, a simple LIST Z command will clear the entire screen and put the cursor at the home position, with Z being equal to any number greater than the highest line number currently in memory. This causes nothing to be listed, so this bit of housekeeping clears the way for a clear screen so that any following output will be uncluttered. At these addresses, do not set the high bit to one as the previous ones were; simply load them as is.

Using Tiny BASIC

To squeeze long programs into small memory areas:

- Use no spaces in the programs. The programs will be difficult to read, but you will save a byte of memory for each space you don't use.
- Use abbreviations; for example, PR for PRINT or variable character for an often-used large number.
- Eliminate inessential words, for example, LET, THEN.

To speed up Tiny BASIC:

- Use variables, which are interpreted faster than numerals.
- Use the word LET. (You must decide whether speed or memory space is more important.)
- Put often-used routines into low memory. Give them the lowest line numbers.

These ideas should help you develop your own techniques to make your programs shorter and easier to write. ■

Printing the North Star Disk Directory From BASIC

Implemented via the mystical majesty of assembly language.

Jan Messersmith
PO Box 224
Brownsburg, IN 46112

Did you ever wish you could print the North Star disk directory from a BASIC program? It might be convenient to see that a file has actually appeared on a disk when debugging a program.

You can usually do this easily by executing a manual two-finger halt (control-C), typing CAT (the direct mode command to list the directory in release 4 NS BASIC) and typing CONT. This will, theoretically, not disturb anything.

As you may have discovered by now, 100 CAT or 100 PRINT CAT or 100 A$ = CAT/ PRINT CAT will not work at all. There doesn't seem to be any easy way of getting program control over directory printing.

A Solution

I have recently discovered the mystical majesty of assembly language. As I was fumbling through my dog-eared NS DOS manual, I found a page that finally made some sense to me. Under the DOS Library Routine Entry Points section is an entry point called List. Its hex address is 2025 and requires the device number (drive number) in the A register (accumulator) when you call it from BASIC.

The hex address is where the entry point lives in memory, and the accumulator is the door that usually passes information to and from the CPU and whatever it is talking to. (In this case, BASIC is talking to a tiny assembly-language program living somewhere else in memory.)

According to the NS BASIC manual, the format of a call instruction is A = CALL (decimal address, parameter). The parameter is optional, but I can use it to pass to the assembly-language program the number of the drive that contains the catalog I wish to examine.

The second argument (the parameter) is passed to the D and E register pair, which will hold two bytes because it is 16 bits wide. Since I will never be concerned about numbers higher than three or four, I only need to be concerned with the contents of the E part of the D and E pair.

Writing the Program

From reading the page in the DOS manual I learned two things: I must place the number of the drive in the accumulator and I must cause the 8080 chip to go to the address where the List routine resides.

If you carefully examined a list of 8080 mnemonics (words that stand for machine operations), you could figure out how to do this. There is an instruction that reads MOV A,E. This means move to register A the contents of register E. Example 1 shows how this looks written out.

```
MOV  A,E  ;MOVE DEVICE NUMBER FROM REGISTER E TO ACCUMULATOR
```

Example 1.

The comment to the right of the semicolon is a remark (in CP/M assembly language, the semicolon is the equivalent of REM in BASIC). This is the first line of the assembly-language program. Now I have to tell the CPU that the next instruction it is to execute is located in RAM at 2025H.

I do not require that any conditions be met (IF in BASIC); I want an unconditional branch to a subroutine (GOSUB in BASIC). The instruction I want is Call. Since the computer will not respond "Number please?", I must tell it where to call.

The next line in the program is shown in Example 2. Here I have a minor problem. The computer likes to have the two bytes of the address fed to it backwards. (Some feeble technical explanations claim, "that's

```
CALL   02025H   ;CALL 02025H, ADDRESS OF DOS 'LIST' ROUTINE
```

Example 2.

```
0000   7B       MOV    A,E      ;MOVE DEVICE $ FROM E TO A
0001   CD2520   CALL   02025H   ;CALL 2025H, ADDRESS OF
                                ;DOS 'LIST' ROUTINE
0004   C9       RET             ;RETURN FROM BASIC 'CALL'
```

Example 3.

the way we wanted it," but these so strain my credulity that I will not discuss them here.)

Now I have sent control to the routine that will list the directory on the screen, or whatever I use for a console device. To use the program, I must return control to BASIC when the portion of DOS I'm using as a subroutine executes its own return. This I do with a simple RET instruction. The program now looks like Example 3. Notice how the assembler automatically reverses the two byte address.

I have added a couple more columns on the left. The first contains the address of each instruction in the program. I chose to put the first address at 0000H, somewhat arbitrarily. You can start with the first of any five consecutive bytes of free RAM that you know is not being used for something else.

The second column contains the actual bytes that will be present at the addresses in the first columns. I got these from a list of hexadecimal equivalents of the mnemonics (of course, some of them are not instructions, but addresses, such as the 2520 backwards). To clarify this further, I've listed the program slightly differently in Example 4.

Now it is easier to see the exact hexadecimal byte sequence and where they go in memory.

I now return to BASIC. By a simple process of filling those spots in memory with the numeric values in the proper sequence, I can create the program in memory and call it from a BASIC program.

First, however, I must convert both the addresses and the values to decimal. Except in the trivial cases we are using here, I heartily endorse the Texas Instruments Programmer calculator, which converts immediately any reasonable number in any of three bases (octal, decimal, hexadecimal) into an equivalent number in one of the other bases.

While I'm at it, I'll write a BASIC subroutine, which I can use anywhere in a program to list the current directory (Example 5).

If I wanted to get fancy, I would enter this routine at line 1000 only on its first use. After the first GOSUB 1000, I could save a fraction of a second each time I needed to see the catalog by entering at line 1060.

This is hardly worth the effort. But the point is that once the values have been filled into their respective memory locations, this process does not have to be repeated. In fact, any line that contains the statement A = CALL(0,N) will generate a directory listing. Instead of a subroutine, I could include the fill statements at the beginning of a program and just use the CALL (0,N) statement.■

```
0000  7B  MOV A,E
0001  CD  CALL
0002  25  25
0003  20  20
0004  C9  RET
```

Example 4.

```
1000 REM *** ROUTINE TO PRINT CURRENT DIRECTORY ***
1010 FILL 0,123
1020 FILL 1,205
1030 FILL 2,37
1040 FILL 3,32
1050 FILL 4,201
1060 A = CALL(0,N) \ REM N IS DRIVE NUMBER
1070 PRINT \ PRINT "CURRENT DIRECTORY ON DRIVE ",N
1080 RETURN
```

Example 5.

Hard Copy For Apple GRAPHICS

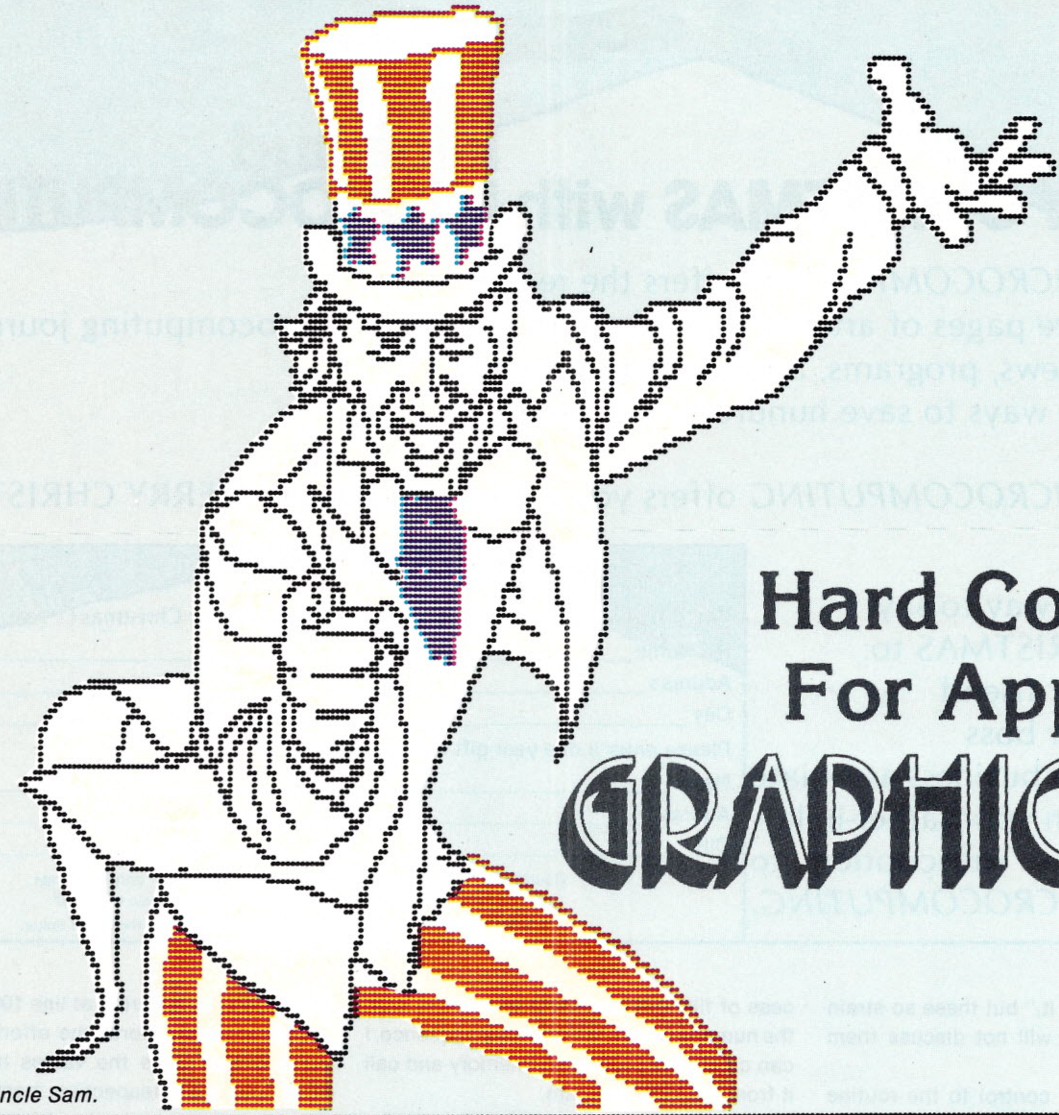


Fig. 1. Uncle Sam.

Thomas D. Brock
Dept. of Bacteriology
University of Wisconsin
Madison, WI 53706

The Apple high-resolution feature makes some fascinating graphics but does not provide an easy way of getting hard copy. I tried photographing the television screen, but this was a little cumbersome, required photographing in a completely dark room and, without considerable darkroom effort, did not give large enough pictures.

I thought there ought to be a way of printing the high-resolution screen, using an incremental printer such as the Diablo. But a brief examination of the memory area where high-resolution pictures are stored—2000 to 3FFF hex (8192 to 16383 decimal) for page 1 or 4000 to 5FFF hex (16384 to 24575 decimal) for page 2—revealed that organization was quite complex and not immediately decipherable.

Fortunately, about the time that I had decided it was not worth the effort, Darrell Smith's article in the September 1979 *Microcomputing* appeared, describing an algorithm for scanning high-resolution memory line-by-line. I was able to use this article to develop a program that printed dot-for-dot a high-resolution picture on my Diablo 1640 printer.

Developing the Program

Although Smith's algorithm makes it possible to scan the high-resolution screen vertically and horizontally line-by-line, you are not home free. There are only 40 horizontal memory locations, and yet 270 dots are plotted horizontally across the screen. How is this accomplished?

Well, each of the 40 bytes contains eight bits, but only seven bits in each byte are used. The eighth is completely ignored. Thus, when evaluating each byte, you must ignore the last bit.

But how about the various colors available in the high-resolution mode? What does the HCOLOR command do?

When HCOLOR is executed, a mask is set at location 00E4 hex (228 decimal). With HCOLOR 3 or 7, each of the first seven bits is set, and you can plot a dot anywhere on the screen. With HCOLOR 0 or 4, each of the first seven bits of the mask is a zero, and no dots are plotted on the screen (thus leading to black). With HCOLOR values of other numbers, some bits are set and others are not; if you carry out an HPLOT routine, you will get a colored dot or line. The color will depend partly upon which dot it is and partly upon your television screen.

The Diablo does not print out in color, so the complications involved here are not worth working through. Since the Apple high-resolution system ignores the high-order bit in any case, it is best to set HCOLOR = 3, because then a byte will read as zero if none of the bits is set by HPLOT.

As you scan a line, you need to read each byte and determine which, if any, of the first seven bits are set. To speed up the scanning of the screen, you first test to see if a byte is zero and skip by it. Since many bytes will be zero, it is considerably faster to print the screen if you ignore these bytes. Lines 185 and 186 accomplish this.

Then, for the bytes that have bits set, you must determine which bits are set. I developed a simple routine that permits analyzing any number less than 256 and

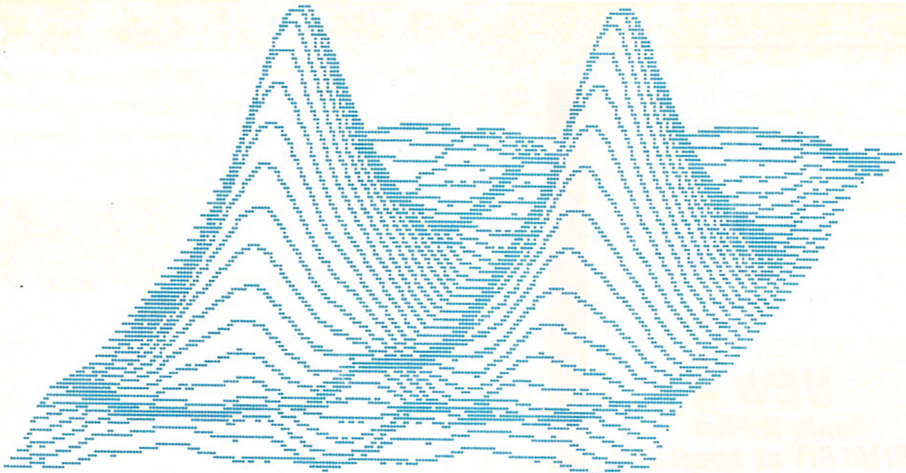


Fig. 2. Double Bessel function.

Horizontal index	Size	Vertical index	Size
CHR$(1,1)	0"		
CHR$(2,2)	2.25"	CHR$(2,2)	4"
CHR$(3,3)	4.5"	CHR$(3,3)	7.99"
CHR$(4,4)	6.75"	CHR$(4,4)	12"
CHR$(5,5)	9"	CHR$(5,5)	22.5"
CHR$(6,6)	11.25"		

Table 1. Picture dimensions with different settings of horizontal and vertical movement index.

printing out its bit pattern. I found this routine useful in studying how HCOLOR and HPLOT work and incorporated it into my program, lines 200 to 250, to print out each bit that is set.

Once you know a bit is set and where on the page the corresponding dot should be printed, you must tell the Diablo to print the dot. The Diablo 1620/1640 printers have two modes that can be used for graphics. One is called graphics mode, and the other is absolute tab. The absolute tab mode is simpler to use.

The Diablo printhead can be instructed to move horizontally in increments as small as 1/120 of an inch, and vertically as small as 1/48 of an inch. The amounts of horizontal and vertical movement are set by the values of CHR$ in lines 80 and 90. The values used in the program give a 9 × 8 inch picture size, almost the format of the high-resolution screen. If the shape of the picture is critical, the horizontal and vertical formats chosen must be proportional to those on the screen. You can select values that will print a highly distorted picture.

However, because the value for CHR$ must be an integer, there is some restriction on the print format. Table 1 gives horizontal and vertical dimensions that the printed picture will take with different values of CHR$.

Once you have initialized the Diablo absolute tab mode, a single space output from the Apple to the Diablo will move the printhead the defined distance, and a single backspace will move the printhead back the same distance. Likewise, a single line feed will move the paper down the defined distance, and a single negative line feed will move the paper up the same distance. You need only to scan each line, print bits as required and go to the next line. With this approach, you don't have to keep track of where the Diablo printhead is vertically, but only horizontally.

Line 270 sends out a carriage return, and line 280 sends out a line feed. In line 440, variable H9 keeps track of the horizontal position of the printhead (it is initialized for each new line in line 160), and lines 400 to 430 tell the printhead how far to move horizontally before printing. Line 450 then prints the dot, an ASCII period, and line 460 backs the printhead to where it had just printed (since each print action in absolute tab mode results in a movement one space to the right).

The Program

The actual program (Listing 1) thus turns out to be surprisingly short. It takes about 15–20 minutes to print an average-sized drawing, using the Integer BASIC program given. An Applesoft program originally written to do the same job took me much longer.

Since Integer BASIC does not have the CHR$ function, which is essential for this program, I introduced the CHR$ routine (line 10) given by Val Golding in Call-Apple. Lines 70 and 75 then define the various ASCII characters required by the Diablo for its absolute tab functions.

Once I wrote the program, I was interested in testing it on one of the high-resolution pictures in the Apple Contributed Library. Figs. 1 and 2 give typical printouts. Note that HCOLOR is set to 3. Also note the page of high-resolution graphics used for the creation of the picture selected in line

```
  1 REM  HIRES PRINT FOR DIABLO 1620
    /40
  2 REM  PRINTS EACH DOT ON THE
  3 REM  HIGH RESOLUTION SCREEN
  4 REM  MAKE PRINT WITH HCOLOR=3
  5 REM  PAGE 1 IS HGR
  6 REM  PAGE 2 IS HGR2
  8 REM  LINE 10 IS INTEGER
    CHR$ FUNCTION
  9 REM  SEE CALL -APPLE SEPT 79
 10 DIM CHR$(126): FOR I=129 TO
    255: POKE 1927+(I-1),I: NEXT
    I: POKE 2182,30
 11 INPUT "PAGE 1 OR 2",D
 12 IF D=1 THEN GOTO 14
 13 START=16384: GOTO 20
 14 START=8192
 15 REM  HGR1 STARTS AT 8192
 16 REM  HGR2 STARTS AT 16384
 20 PR#3: PRINT " ";
 21 REM  LINE 20 INITIALIZES
    PRINTER
 30 C9$=".":SP$=" "
 31 REM  C9$ IS THE CHARACTER
    PRINTED
 70 E$=CHR$(27,27):US$=CHR$(31,
    31):RS$=CHR$(30,30):HT$=CHR$
    (9,9):LF$=CHR$(10,10):BS$=CHR$
    (8,8)
 75 AS$=CHR$(1,1):AT$=CHR$(2,2)
    :VT$=CHR$(11,11)
 76 REM  LINES 70-75 DEFINE
    VARIOUS ASCII CODES FOR DIABLO
 80 PRINT E$;US$;CHR$(5,5);
 81 REM  LINE 80 SETS HORIZ PRINT
    MOVEMENT
 90 PRINT E$;RS$;CHR$(3,3);
 91 REM  LINE 90 SETS VERT PRINT
    MOVEMENT
 95 PRINT E$;HT$;AS$;LF$;
 96 REM  LINE 95 IS CR/LF;HOMES
    PRINT-HEAD
 97 REM  LINES 100-150 SCAN IN
    VERT DIRECTION
100 FOR Y=0 TO 191
110 A=Y/64
120 Y1=Y MOD 64
130 B=Y1/8
140 C=Y1 MOD 8
150 P=START+(A*40)+(B*128)+(C*1024)
152 REM  LINES 160-260 SCAN IN
    HORIZ DIRECTION AND PRINT
160 X9=0:H9=0
170 FOR X=0 TO 39
180 R= PEEK (P+X)
185 IF R=0 THEN X9=X9+7
186 IF R=0 THEN GOTO 260
187 REM  LINES 185-186 SKIP
    BLANK BYTES
190 I=0
191 REM  LINES 200-250 CHECK
    EACH BIT TO SEE IF SET
200 R3=R MOD 2
210 IF R3<>0 THEN GOSUB 400
220 X9=X9+1
230 R=R/2
240 I=I+1
250 IF I<7 THEN GOTO 200
260 NEXT X
270 PRINT E$;HT$;AS$;
280 PRINT LF$;
281 REM  LINES 270-280 RETURN
    CARRIAGE FOR NEXT LINE
290 NEXT Y
300 END
398 REM  LINES 400-470 ARE THE
    PRINT ROUTINE
400 N9=X9-H9
405 IF N9=0 THEN RETURN
410 FOR J9=1 TO N9
420 PRINT SP$;
430 NEXT J9
440 H9=H9+N9
441 REM  H9 KEEPS TRACK OF
    HORIZ PRINT POSITION
450 PRINT C9$;
460 PRINT BS$;
461 REM  AFTER PRINTING CHAR
    BACKSPACE TO KEEP POSITION RIGHT

470 RETURN
```

Listing 1. Program in Integer BASIC to print the Apple high-resolution screen.

11. I noticed a problem with the Integer BASIC CHR$ function when printing some high-resolution pictures. I corrected this by loading the picture, resetting and rebooting the Apple and then loading my print program and running it.

Conclusion

Now that I can print the high-resolution screen, I am thinking of a wide range of possibilities for using this capability. With the use of the high-resolution character generator and character table in Apple Contributed Library, Vol. 3, I can print upper and lowercase characters anywhere on the high-resolution screen. Thus, I can draw pictures and graphs, label them, change them in any way desired, print them out and get hard copy. The copy is suitable for reproduction and is much easier (and more fun) than photographing the television screen.

Although the routines I have given are designed for the Diablo, there is no reason why similar routines could not be devised for any printer capable of adjustable spacing of the printhead. Dot matrix printers make possible the printing of seven bits at once, and consequently work considerably faster, but the quality of the print from the Diablo is unsurpassed. This routine greatly extends the capability of the Apple for high-resolution graphics. ■

How's your love life?

A little dull around the edges? Routine? Predictable? Boring? Maybe all it needs is a little Interlude. Interlude is the most stimulating computer game ever conceived. It combines a computer interview, an innovative programming concept, and a one-of-a-kind manual to turn your love life into exciting, adventurous, delicious fun!

Interlude is: romantic...playful...outrageous...a fantasy. Interlude is: ■ A Bed of Roses (Interlude #1) ■ Mata Hari (Interlude #49) ■ The Chase (Interlude #7) ■ Rodeo! (Interlude #71) ■ The King and I (Interlude #60) ■ Some Enchanted Evening (Interlude #84) ■ Caveman Caper (Interlude #82) ■ From Here to Ecstasy (Interlude No. 30) ■ Satin Dreams (Interlude #72).

More than 100 Interludes are included in the program. Most are described in detail in the accompanying manual, but several surprise Interludes are buried in the program awaiting that very special time when your interview says you're ready. (When you learn secret Interlude #99, your love life may never again be the same!) Interlude can give you experiences you'll never forget. Are you ready for it?

Interlude™
The Ultimate Experience. ✓235

INTERLUDE, 10428 Westpark, Houston, Texas 77042. I'm really ready. Send my Interlude today.

Apple II (16K)*
☐ Cassette ($16.95)
☐ Diskette ($19.95)
☐ Diskette—Pascal or DOS 3.3 ($19.95)
Add $1.50 for shipping and handling.

TRS-80 (Level II-16K)**
☐ Cassette ($16.95)
☐ Diskette ($19.95)

Poster
☐ 20"x 24" reproduction of this ad without ad copy ($4.95—includes shipping charges)

Available for immediate shipment.

Please enclose your check payable to INTERLUDE or complete the charge information:

☐ MASTERCARD ☐ VISA All charge customers must sign here. ____________

Account No. ____________ Expiration date ________ MasterCard Bank Code ________

CHARGE CUSTOMERS: Order by phone toll-free! **1-800-231-5768 Ext. 306** (Texas: 1-800-392-2348 Ext. 306)

Name ____________ Age ______

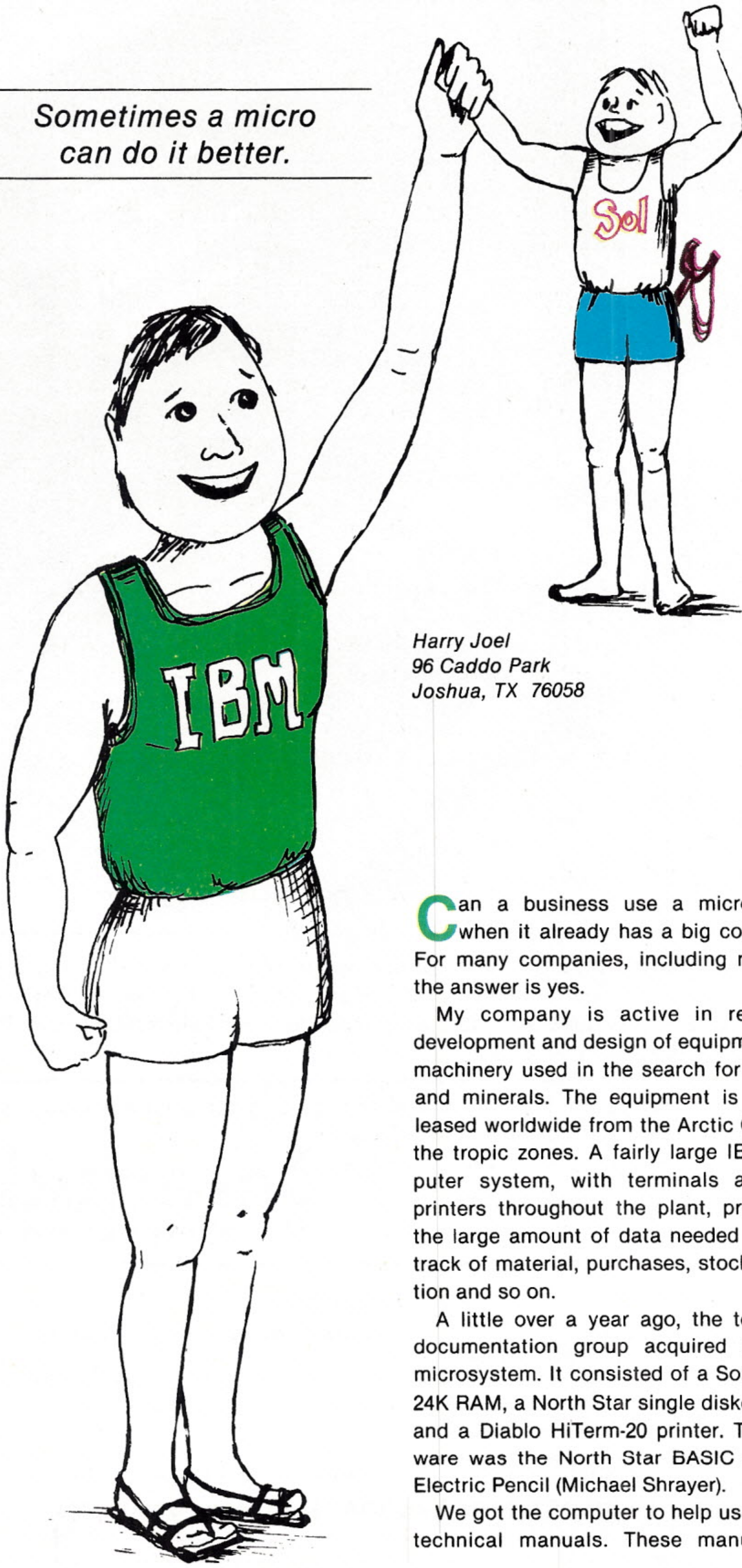
Address ____________

City ____________ State ________ Zip ________

*Apple II is a registered trademark of Apple Computers, Inc. **TRS-80 is a registered trademark of Radio Shack, a Tandy Co.

David and Goliath

Sometimes a micro can do it better.



Harry Joel
96 Caddo Park
Joshua, TX 76058

Can a business use a microsystem when it already has a big computer? For many companies, including my own, the answer is yes.

My company is active in research, development and design of equipment and machinery used in the search for oil, gas and minerals. The equipment is sold or leased worldwide from the Arctic Circle to the tropic zones. A fairly large IBM computer system, with terminals and line printers throughout the plant, processes the large amount of data needed to keep track of material, purchases, stock allocation and so on.

A little over a year ago, the technical documentation group acquired a small microsystem. It consisted of a Sol-20 with 24K RAM, a North Star single diskette unit and a Diablo HiTerm-20 printer. The software was the North Star BASIC and the Electric Pencil (Michael Shrayer).

We got the computer to help us prepare technical manuals. These manuals go through several typing/proofreading cycles and often must be retyped a number of times. The micro-based word-processing system effectively eliminates this repetition.

Copy is typed into the system. Changes are quickly and effortlessly done on the video screen. Old material is called back from disk, and the edited work is sent back to the disk after a final copy is printed at high speed on the terminal.

Our typists learned to use the equipment quickly. Once they had mastered a few new routines required by the word processor equipment, their work took less time and was more professional.

Since we got the computer, we've processed about 20 manuals. The copy is still available on disk and back-up disk for later changes. We also have boilerplates on disk—text that may be repeated in a number of publications.

Overall, we now have faster turnaround, better contents and style and happier employees.

If this system had done nothing but serve as an efficient word processor, it probably would have paid for itself by now. But other applications soon developed.

While all this word processing was going on, employees from other departments became curious. They wanted to know what else this "computer typewriter," as most called it, could do.

Well, it certainly could be taught to do specialized, small data processing and form-handling chores when properly programmed in BASIC. The chief caretaker of the system had learned how to program in BASIC and was waiting for some challenging opportunities to make the system work even harder. It did not take long before the jobs came in.

As you will see from the examples, none of these jobs could be effectively done on the large system. The large system is rigidly designed to do specific jobs. It is much faster and has a larger data capacity. With small and special jobs, it is not feasible to use the big system. Even more important, the required hard copy is not available in the format we needed.

The small system is decidedly better

when it comes to total turn-around time.

Some Examples

Example 1. We bought a small company. Their drawing number system had to be converted to our system.

To make matters more interesting, the drawing number for subassemblies and assembly drawings was an alpha code. The code book listed entries starting at A through Z, then AA through AZ and so on. Without the small system, a typist would have had to tediously prepare a cross-index, carefully indexing from, for instance, DKZ to DLA while typing in the corresponding ten-digit code for the new drawing number.

We wrote a short BASIC program, debugged it and printed out the 80-page list within about four hours. The same program, slightly edited, then printed out a set of transparent labels with the new numbers. These labels were attached to all prints by the drafting department. A copy of the cross-index went to departments that needed this information.

Example 2. Our printroom keeps a master log (several three-ring binders) for drawing numbers, titles and drawing size. As new products are developed, new blocks of drawing numbers must be incorporated in this system. We wrote another short BASIC program to print a sequence of ten-digit drawing numbers along the left margin of a logbook page. The spacing was four lines per inch instead of the standard six lines per inch, but the Diablo printer can easily be set up for this or any other spacing.

Example 3. We installed a new Engineering Change Notice procedure for plantwide application. Initially, we logged all issued ECOs into a handwritten logbook. After about 2500 entries it became clear that an engineer preparing a new ECO needed to know the past ECO history on a particular item. The manual search through the logbook was not only too slow, but also not always correct. An item could easily be overlooked.

We therefore developed a BASIC program that let us enter the drawing number/ECO record in a master disk file, correct any entry, search by drawing number for all ECOs issued, selectively search by start and end number ECO, selectively print out and, as an add-on, search for all ECOs written for all subassemblies within a given end item. The drawing number system allowed this combination search because all piece parts and subassemblies designed for a particular end item contained an identical four-digit code within the part number.

Due to its complexity, the program was carefully designed around functional modules. It incorporated convenience features for the operator (prompts, automatic execution on start-up, free disk space information) and input error detection.

We exercised the program for two weeks, and found a few more hang-ups that could have caused much grief later. After final debugging, we now have a useful and efficient tool.

Example 4. We developed another simple routine for making additional text entries in the drawing logbook. Record entries are done one line at a time.

The typist enters a line on the video terminal, proofs the text while it is still on the screen, makes any required corrections and hits the enter key. The tiny BASIC program turns on the printer, prints the records in proper tabulated form and returns control to the display. It's not impressive programming, but the typist loves it.

Example 5. Up-to-date parts lists are an important part of any design, manufacturing and purchasing activity. With the ever-increasing line of products, we needed a simple procedure that lets us create new lists, update existing lists (see ECO activity above) and rapidly distribute the parts documents.

We had to meet three specific requirements. First, for printroom use, the lists had to be printed on vellum and have good print quality. Second, corrections had to be made with minimum fuss and on short notice. Third, about 5000 lists had to be put into the system immediately, and at least 10,000 lists ultimately had to be kept on the data base for quick retrieval during updates.

The original system did not have the required disk storage capacity, and the tech-writing group was already using the system for the better part of each working day. Thus, we bought another Sol-20 with a dual disk Helios drive and Diablo printer.

Here is the simple routine that works for us: A copy of the Helios system disk contains a macro command file. After startup the disk signs on, the number of free sectors is displayed, the Electric Pencil loads into memory, and the data disk in the second disk drive is activated.

The operator then enters the parts list content and saves it under its own file name. The blank vellum preprinted form is put in the printer. A yellow carbon against the backside of the vellum provides improved print quality in the Diablo machine.

Because of the format requirements of the Electric Pencil, the form is actually printed in two fields. After the left side printing is complete, a "rollup" command on the text file returns the form to line one. Another print command, again part of the text file, moves the left margin over and the remaining half of the form is printed.

We make a backup copy of all disks so never more than half a day's worth of work is lost by equipment or power failures. On the average, each eight-inch disk holds 200 complete parts lists. So far, we have processed nearly 3000 lists, with many updates already done on a large portion of this total. The disk number and file number is printed on every document, so retrieval is easy. We now have clean, readable and correct documents no matter how many times changes are made.

With small and special jobs, it is not feasible to use the big system . . . The small system is decidedly better when it comes to total turn-around time.

Durability

How has the equipment held up? Both systems are on line about seven hours during each working day. Disk activity is much higher than would reasonably be expected in a home compter. With commonsense care in disk handling and storage, we have had excellent reliability from our data disks. The hardware reliability is also quite good. One keyboard had a worn out return key pad, which we repaired in-house.

After the third month, the North Star disk system acted up. A careful check disclosed a partially made connection on the flat cable crimp connector on the disk controller. The Helios system went down once due to a deteriorating head position servo amplifier. The local store (Computer Port in Arlington, TX) quickly put the system back in order.

In summary, the microprocessor systems at our company have proven their worth, have earned their keep and will inspire us to even better applications. If your boss needs a little convincing, we hope this story will give you some leverage to get a good system for your company. ■

INSTANT "PET" FOOD

Available in the following varieties: GAMES, FINANCE, HOME USE, SIMULATIONS and ELECTRONICS. We now have these 23 flavors on hand to tempt your PET's palate. Study these pages for the tastiest software yet—guaranteed to please your PET* or CBM*...

PENNY ARCADE Enjoy this fun-filled package that's as much fun as a real penny arcade—at a fraction of the cost!
•**Poetry**—Compose free verse poetry on your computer.
•**Trap**—Control two moving lines at once and test your coordination.
•**Poker**—Play five-card draw poker and let your PET deal and keep score.
•**Solitaire**—Don't bother to deal, let your PET handle the cards in this "old favorite" card game.
•**Eat-Em-Ups**—Find out how many stars your Gobbler can eat up before the game is over.
These six programs require the PET with 8K. **Order No. 0044P $7.95.**

DUNGEON OF DEATH Battle evil demons, cast magic spells, and accumulate great wealth as you search for the Holy Grail. You'll have to descend into the Dungeon of Death and grope through the suffocating darkness. If you survive, glory and treasure are yours. For the PET 8K. **Order No. 0064P $7.95.**

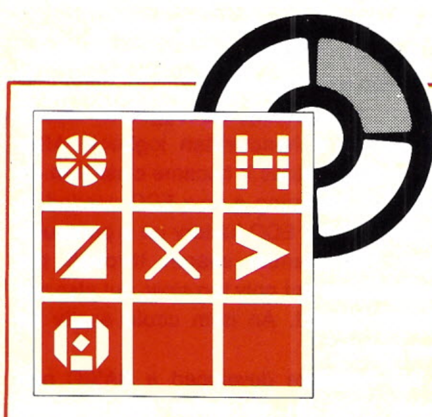
QUBIC-4/GO-MOKU Play two ancient games on your modern PET. The two programs included are:
•**Qubic-4**—Play a multi-dimensioned game of tic-tac-toe.
•**Go-Moku**—Line up five of your men while blocking the PET's moves.
These one-player games require 8K of memory. **Order No. 0038P $7.95.**

PET DEMO I You can give yourself, your family, and your friends hours of fun and excitement with this gem of a package.
•**Slot Machine**—You won't be able to resist the enticing messages from this computerized one-armed bandit.
•**Chase**—You must find the black piece as you search through the ever-changing maze.
•**Flying Pheasant**—Try to shoot the flying pheasant on the wing.
•**Sitting Ducks**—Try to get your archer to shoot as many ducks as possible for a high score.
•**Craps**—It's Snake Eyes, Little Joe, or Boxcars as you roll the dice and try to make your point.
•**Gran Prix 2001**—Drivers with experience ranging from novice to professional will enjoy this multi-leveled race game.
•**Fox and Hounds**—It's you against the computer as your four hounds try to capture the computer's fox.
For true excitement, you'll need a PET 8K. **Order No. 0035P $7.95.**

CODE NAME:CIPHER

Enjoy that same feeling of intrigue and discovery with the Code Name: Cipher package. Included are:
•**Memory Game**—Would you like to match your memory against the computer's? You can with the Memory Game.
•**Codemaster**—One player types in a word, phrase, or sentence, and the PET translates that message into a cryptogram. The other player must break the code and solve the cryptogram in the shortest time possible.
•**Deceitful Mindmaster**—This isn't your ordinary Mastermind-type game. You must guess the five letters in the hidden code word.
•**Code Breaker**—Cracking this code won't be as easy as cracking walnuts. You'll need to flex your mental muscles to win this game.

If you want a mental challenge, then Code Name: Cipher is for you. For the 8K PET. **Order No. 0112P. $7.95.**



MIMIC Test your memory and reflexes with the five different versions of this game. You must match the sequence and location of signals displayed by your PET. This one-player program includes optional sound effects with the PET 8K. **Order No. 0039P $7.95.**

*A trademark of Commodore Business Machines

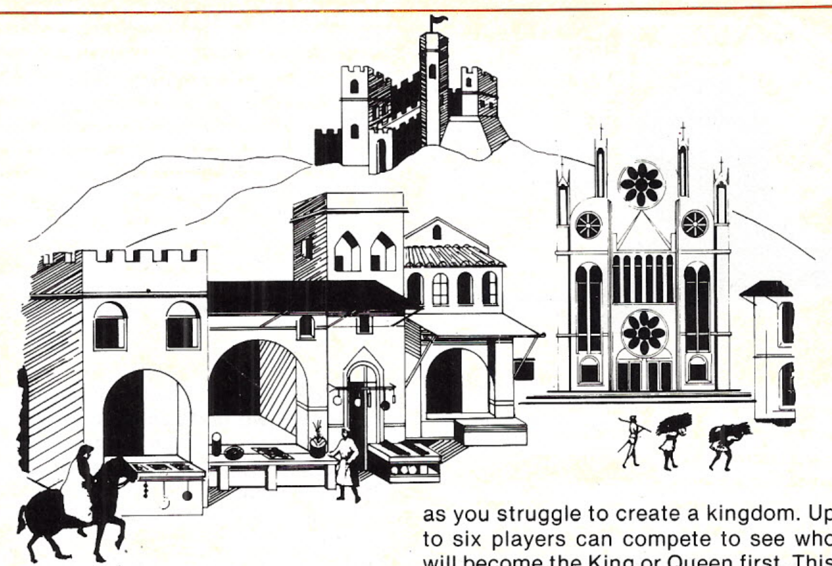
Instant Software™

Prices subject to change without notice.

PETERBOROUGH, N.H. 03458
603-924-7296

CHIMERA If you think the legendary Chimera was hard to handle, wait until you try the Chimera package. Included are:

- **Reflex**—Round and round the little white ball rolls. Only fast reflexes can guide it into the center of the maze.
- **Dragon**—You'll have to shoot down those pesky, fire-breathing dragons with your cannon. If you succeed your castle will be safe, if not it will mean a call to your fire insurance company. For one player.
- **Dungeon**—A very punctual guard comes down to the dungeon every day to torture you. This means that you have only thirty seconds to dig your way under the castle to freedom. For one player.
- **Dragon Hunt**—You must go forth and slay a fire-breathing dragon. The only thing that will protect you from the flames is your shield, if you know when to use it. For one player.
- **Dropoff**—You must make your opponent's men "dropoff" the board by moving and firing your own men. For one or two players. **Order No. 0110P. $9.95.**



SANTA PARAVIA AND FIUMACCIO Become the ruler of a medieval city-state as you struggle to create a kingdom. Up to six players can compete to see who will become the King or Queen first. This program requires a PET 16K. **Order No. 0175P.$9.95.**

BASEBALL MANAGER This pair of programs will let you keep statistics on each of your players. Obtain batting, on-base, and fielding averages at the touch of a finger. Data can be easily stored on cassette tape for later comparison. All you need is a PET with 8K. **Order No. 0062P $14.95.**

TANGLE/SUPERTRAP These two programs require fast reflexes and a good eye for angles:

- **Tangle**—Make your opponent crash his line into an obstacle.
- **Supertrap**—This program is an advanced version of Tangle with many user control options.

Enjoy these exciting and graphically beautiful programs. For one or two players with an 8K PET. **Order No. 0029P $7.95.**

ELECTRONIC ENGINEER'S ASSISTANT Now you can use your computer to analyze designs for filter circuits and microstrip transmission lines.

- **Network Analysis**—Your computer can help you design and analyze four-terminal AC networks. Just enter the input load impedance, component values, and the frequency range. Your computer will analyze the circuit and display the gain, the real input impedance, and the imaginary input impedance throughout the entire frequency range. There's even an optional plotting routine for graphing frequency response.
- **Microstrip**—This program can help you design microstrip transmission lines for printed circuit boards and other mediums. You can get either the dimensionless width-to-height ratio of the supporting medium or the impedance of the system. For the PET 8K. **Order No. 0085P $9.95.**

CASINO I These two programs are so good, you can use them to check out and debug your own gambling system!

- **Roulette**—Pick your number and place your bet with the computer version of this casino game. For one player.
- **Blackjack**—Try out this version of the popular card game before you go out and risk your money on your own "surefire" system. For one player.

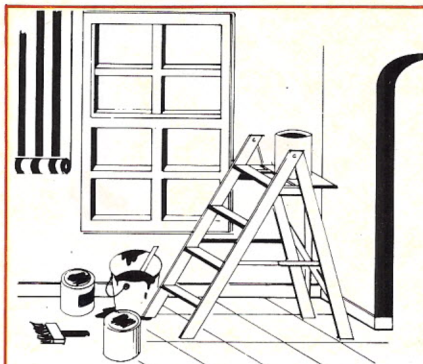
This package requires a PET with 8K. **Order No. 0014P $7.95.**

CASINO II This craps program is so good, it's the next best thing to being in Las Vegas or Atlantic City. It will not only play the game with you, but will also teach you how to play the odds and make the best bets. A one-player game, it requires a PET 8K. **Order No. 0015P $7.95.**

ARCADE I This package combines an exciting outdoor sport with one of America's most popular indoor sports:

- **Kite Fight**—It's a national sport in India. After you and a friend have spent several hours maneuvering your kites across the screen of your PET, you'll know why!
- **Pinball**—By far the finest use of the PET's exceptional graphics capabilities we've ever seen, and a heck of a lot of fun to boot.

Requires an 8K PET. **Order No. 0074P $7.95.**

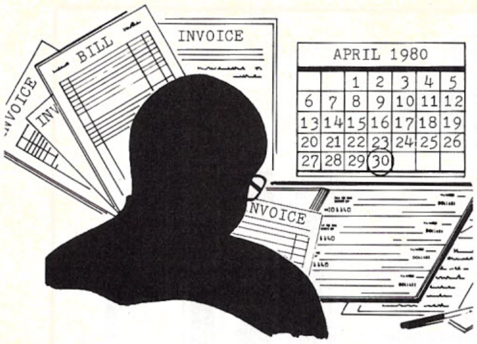


DECORATOR'S ASSISTANT This integrated set of five programs will compute the amount of materials needed to redecorate any room, and their cost. All you do is enter the room dimensions, the number of windows and doors, and the base cost of the materials. These programs can handle wallpaper, paint, panelling, and carpeting, letting you compare the cost of different finishing materials. All you'll need is a PET 8K. **Order No. 0104P $7.95.**

Instant Software™

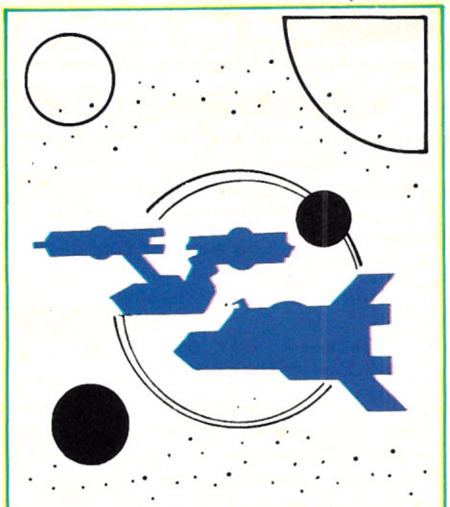
Prices subject to change without notice.

PETERBOROUGH, N.H. 03458
603-924-7296



MORTGAGE WITH PREPAYMENT OPTION/FINANCIER These two programs will more than pay for themselves if you mortgage a home or make investments:
- **Mortgage with Prepayment Option** – Calculate mortgage payment schedules and save money with prepayments.
- **Financier** – Calculate which investment will pay you the most, figure annual depreciation, and compute the cost of borrowing, *easily and quickly*.

All you need to become a financial wizard is an 8K PET. **Order No. 0006P $7.95.**



TREK-X Command the Enterprise as you scour the quadrant for enemy warships. This package not only has superb graphics, but also includes programming for optional sound effects. A one-player game for the PET 8K. **Order No. 0032P $7.95.**

ACCOUNTING ASSISTANT This package will help any businessman solve many of those day-to-day financial problems. Included are:
- **Loan Amortization Schedule** – This program will give you a complete breakdown of any loan or investment.
- **Depreciation Schedule** – You can get a depreciation schedule using any one of the following methods: straight line, sum of years-digits, declining balance, units of production, or machine hours.

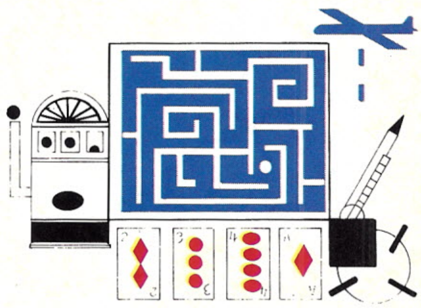
This package requires the PET 8K. **Order No. 0048P $7.95.**

ARCADE II One challenging memory game and two fast-paced action games make this one package the whole family will enjoy for some time to come. Package includes:
- **UFO** – Catch the elusive UFO before it hits the ground!
- **Hit** – Better than a skeet shoot. The target remains stationary, but you're moving all over the place.
- **Blockade** – A two-player game that combines strategy and fast reflexes.

Requires an 8K PET. **Order No. 0045P $7.95.**

TURF AND TARGET Whether on the field or in the air, you'll have fun with the Turf and Target package. Included are:
- **Quarterback** – You're the quarterback as you try to get the pigskin over the goal line. You can pass, punt, hand off, and see the result of your play with the PET's superb graphics.
- **Soccer II** – Play the fast-action game of soccer with four playing options. The computer can play itself or a single player; two can play with computer assistance; or two can play without help.
- **Shoot** – You're the hunter as you try to shoot the bird out of the air. The PET will keep score.
- **Target** – Use the numeric keypad to shoot your puck into the home position as fast as you can.

To run and score, all you'll need is a PET with 8K. **Order No. 0097P $7.95.**



HOOPTEDOODLE

This package is a collection of eight entertaining programs for you and your 8K PET. You'll escape from a monster in an unseen maze, try your luck with the one-armed bandits, cross a treacherous mine field, deflect the "bouncing ball", direct a low level bombing mission, maneuver a high-speed "worm" to score points, launch ground to air missiles, and play a challenging card game.

Having fun with this package is as easy as pressing PLAY on the Recorder.
Order No. 0091P $9.95.

Most of the programs in this catalog were written for the old ROM. They will run in the new ROM correctly if a few minor changes are made.

#0015P-CRAPS: In line 96 insert a cursor control CLR ♥ after the quotation marks (") and before the text BYE, HAVE A NICE DAY!

 # 0022P-CHECKERS: In line 1410 delete the ending semicolon (;). In line 236 delete the cursor control character after the first quote ("). Redo line 4020 so it reads 4020 PRINT:PRINT

BACCARAT: In lines 360 and 480 add a blank either before or after the text in quotes.

 # 0038P-QUBIC-4: In all places where POKE 525 and WAIT 525 are used change them to POKE 158 and WAIT 158.

 # 0045P-UFO: Line 1220 needs a semicolon (;) added to the end of it.

 # 0104P-DECORATOR'S ASSISTANT: These POKEs should be changed; 519 to 249; 525 to 158; 526 to 159; 527 to 160.

 # 0112P-DECEITFUL MASTERMIND: Add this line; 1675 PRINT

TO ORDER: Look for these programs at the dealer nearest you (see list on the next page). If your store doesn't stock Instant Software send your order with payment to: Instant Software, Order Dept., Peterborough, N.H. 03458 (Add $1.00 for handling) or call toll-free 1-800-258-5473 (VISA, MC and AMEX accepted).

Instant Software™

Prices subject to change without notice.

PETERBOROUGH, N.H. 03458
603-924-7296

Finance and Investment

Attention all would-be millionaires. Now, keep track of your investments by harnessing the power of your Apple II (or Apple II Plus) with the speed of floppy disk storage. The Finance and Investment package has been fashioned to help you, the businessman, to solve some of those time-consuming tasks you face daily. The programs included are:

Loan Amortization Schedule—This program will calculate a complete monthly breakdown of any loan or investment. All you do is enter the amount of the principal, the interest rate, the term of the loan or investment and the number of payments per year. You'll see a month-by-month list of the principal, interest, total amount paid and the remaining balance. Any of the amounts can be listed on a paid-to-date basis, at your option.

Depreciation Schedule—It will compute a depreciation schedule using any one of the following methods: Straight Line, Sum of Years-Digits, Declining Balance, Units of Production or Machine Hours. Just enter data in response to the computer's prompts and you'll see a list of how long the item has been or will be in use, the annual depreciation, the accumulated depreciation and the remaining book value.

Mortgage with Prepayment—Use this program to develop a prepayment plan that will provide optimum savings on the cost of the mortgage, reduce the terms of the mortgage and help avoid overtaxing your income in the process. It will calculate the cost of the original mortgage, as well as the cost and savings on a mortgage with annual prepayments. If you must borrow money to make the prepayments, the computer takes the added interest into consideration.

Financier—This program is designed to take the extensive paperwork out of your daily financial planning. It performs ten common financial calculations that can help you: (1) design optimum investment schedules; (2) check on depreciation rates, amounts and resale values; and (3) let you know exactly what a given loan is going to cost in terms of time and money.

Minimum system requirements are an Apple II or Apple II Plus with 32K of memory, one mini-disk drive and Applesoft BASIC.

Order No. 0162AD $19.95

Skybombers II

Two countries, separated by The Big Green Mountain, are at war. Both nations are equipped with only one means of attack—SKYBOMBERS!

You and your opponent, each representing the nations at war, command opposing fleets of fighter-bombers armed with bombs and missiles. As enemy commanders, each of you has specific orders: Fly across that mountain and bomb the enemy blockhouse into oblivion!

Flying over that innocent looking mountain is not easy for either air force. The aircraft can fire missiles at each other; if that fails, they can ram each other. Sometimes, aircraft encounter falling bombs and are blown to pieces in flight. Desperate pilots can even crash into the enemy blockhouse.

Flight personnel are sometimes forced to parachute from badly damaged aircraft. As they float slowly to earth, they become helpless targets for the enemy to destroy in mid-air.

The sounds of battle, from exploding bombs to the screams from wounded parachutists being attacked, are there to remind each commander of his grim responsibility.

Explosions are graphically displayed for both commanders. The scores for both countries are constantly updated at the bottom of the display screen.

Flying these missions develops into a gripping fascination. Air warfare becomes a vivid reality, as you both play the deadly game of Skybombers II.

The Skybombers II program requires 32K RAM, one disk drive, Applesoft in ROM and the game paddles.

Order No. 0271AD (disk-based version) $19.95.

OTHER PROGRAMS FROM INSTANT SOFTWARE

TRS-80* LEVEL I & II

0001R	Basic and Intermediate Lunar Lander	$7.95
0002R	Space Trek II	$7.95
0004R	Beginner's Backgammon/Keno	$7.95
0007R	Ham Package I	$7.95
0008R	Electronics I	$7.95
0009R	Golf/Cross-Out	$7.95
0017R	Air Flight Simulation	$7.95
0019R	Business Package IV	$9.95
0023R	Oil Tycoon	$7.95
0033R	Bowling	$7.95
0043R	Santa Paravia and Fiumaccio	$7.95
0046R	Othello	$9.95
0050R	Grade Book	$9.95
0057R	Chessmate-80	$19.95
0099R	Typing Teacher	$9.95

TRS-80* LEVEL II

0028R	Ramrom Patrol/Tie Figher/Klingon Capture	$7.95
0034R	Space Trek IV	$7.95
0047R	Who-Dun-It?	$7.95
0049R	Demo II	$7.95
0051R	Ball Turret Gunner	$9.95
0055R	Demo III	$7.95
0056R	Bowling League Statistics System	$24.95
0058R	Programmer's Converter	$9.95
0063R	Cards	$7.95
0065R	Teacher	$9.95
0066R	Mimic	$7.95
0068R	Your Cribbage and Checkers Partner	$9.95
0069R	Household Accountant	$7.95
0070R	Skirmish-80	$9.95
0072R	Financial Assistant	$7.95
0076R	TRS-80* Utility II	$7.95
0077R	Enhanced BASIC	$24.95
0081R	TRS-80* Utility I	$7.95
0082R	Daredevil	$9.95
0084R	Music Master	$7.95
0089R	Energy Audit	$49.95
0092R	Archimede's Apprentice	$9.95
0100R	Video Speed-Reading Trainer	$9.95
0103R	Personal Bill Paying	$7.95
0106R	Airmail Pilot	$7.95
0109R	Body Buddy	$9.95
0111R	Wordwatch	$7.95
0117R	Night Flight	$9.95
0118R	Mind Warp	$9.95
0124R	Winner's Delight	$9.95
0125R	Investor's Paradise	$9.95
0126R	The Communicator	$9.95
0127R	Surveyor's Apprentice	$9.95
0129R	The Wordslinger	$29.95
0130R	Terminal-80	$39.95
0131R	Sales Analysis	$24.95
0132R	Energy Consumption	$9.95
0135R	Executive Expense Report Generator	$9.95
0136R	Beginner's Russian	$9.95
0137R	Everyday Russian	$9.95
0140R	Oracle-80	$75.00
0141R	Battleground	$9.95
0156R	Money Madness	$9.95
0157R	IQ Test	$9.95
0159R	Jet Fighter Pilot	$14.95
0171R	Flight Path	$9.95
0203R	BASIC Programming Assistant	$14.95
0223R	Cosmic Patrol	$14.95
0230R	TLDIS	$14.95
0232R	The Disassembler	$9.95
0250R	IRV	$24.95
5002R	Basic Math Program from EMSI	$80.00

TRS-80* DISKS

0052RD	Energy Audit	$75.00
0075RD	Accounts Payable/Receivable	$199.95
0095RD	Bowling League Secretary	$49.95
0123RD	The One-D Mailing List	$24.95
0139RD	Disk-Scope	$19.95
0147RD	Check Management System	$39.95
0151RD	QSL Manager	$19.95
0152RD	Oracle-80	$99.95
0180RD	Disk Editor	$39.95
0212RD	The Russian Disk	$24.95
0214RD	Teacher's Aide	$39.95
0231RD	DLDIS	$19.95
5000RD	Mail List from Galactic (Mod. I)	$99.00
5001RD	Mail List from Galactic (Mod. II)	$199.00

PET**

0005P	Personal Weight Control/Biorhythms	$7.95
0006P	Mortgage w/Prepayment Option/Financier	$7.95
0014P	Casino I	$7.95
0015P	Casino II	$7.95
0026P	Dow Jone$	$7.95
0029P	Tangle/Supertrap	$7.95
0032P	Trek-X	$7.95
0035P	PET Demo I	$7.95
0038P	Qubic-4/Go-Moku	$7.95
0039P	Mimic	$7.95
0044P	Penny Arcade	$7.95
0045P	Arcade II	$7.95
0048P	Accounting Assistant	$7.95
0054P	Ham Package I	$7.95
0062P	Baseball Manager	$14.95
0064P	Dungeon of Death	$7.95
0074P	Arcade I	$7.95
0083P	Digital Clock	$7.95
0085P	Electronics Engineer's Assistant	$9.95
0091P	Hooptedoodle	$9.95
0097P	Turf and Target	$7.95
0104P	Decorator's Assistant	$7.95
0105P	PET Utility I	$9.95
0110P	Chimera	$7.95
0112P	Code Name: Cipher	$7.95
0175P	Santa Paravia and Fiumaccio	$9.95

APPLE***

0018A	Golf	$7.95
0025A	Mimic	$7.95
0040A	Bowling/Trilogy	$7.95
0073A	Math Tutor I	$7.95
0079A	Oil Tycoon	$9.95
0080A	Sahara Warriors	$7.95
0088A	Accounting Assistant	$7.95
0094A	Mortgage w/Prepayment Option/Financier	$7.95
0096A	Space Wars	$7.95
0098A	Math Tutor II	$7.95
0148A	Air Flight Simulation	$9.95
0174A	Santa Paravia and Fiumaccio	$9.95

APPLE***

0160AD	Math Fun	$19.95
0161AD	Apple Fun	$19.95
0162AD	Finance and Investment	$19.95
0163AD	Paddle Fun	$19.95

WRITE FOR OUR NEW CATALOG!

HEATH****

0087H	Mental Gymnastics	$7.95

*A trademark of Tandy Corporation
**A trademark of Commodore Business Machines
***A trademark of Apple Computer Inc.
****A trademark of HEATH Company

Instant Software™

Prices subject to change without notice.

PETERBOROUGH, N.H. 03458
603-924-7296

✓40

The TC-8 Cassette Interface System

TRS-80 owners: save and load five times faster.

Sherman P. Wantz
424 NW Lakeview Drive
Sebring, FL 33870

The TC-8 cassette recorder interface is one more way to save and load BASIC and machine-language programs on a TRS-80 Model I. The price is unbeatable—$90 for the kit.

The TC-8's chief advantage is that it lets you transfer your programs to and from cassette tape at least five times faster than the TRS-80 Model I system allows. JPC Products Company, manufacturer of the TC-8, claims that its system will record and play back data at 3000 baud, versus 500 baud claimed by Radio Shack.

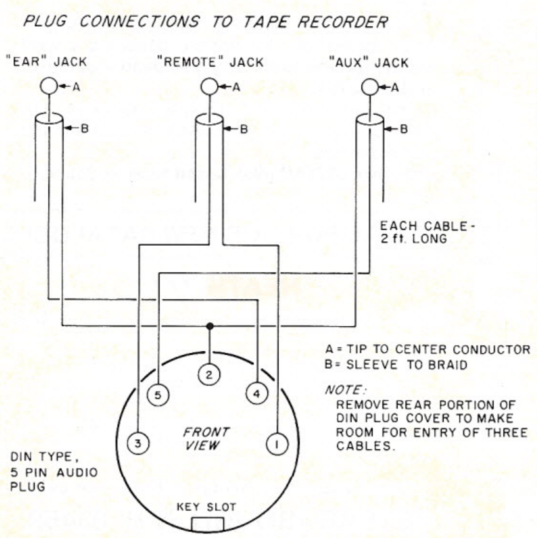
My own measurements show the speed of the TC-8. With my recorder connected directly to my TRS-80, it took nine minutes, 33 seconds to CLOAD the fourth program on one of my cassettes. With my TC-8, it took just one minute, ten seconds, eight times faster.

Certainly, the dead space I had left between programs on the original tape was significant. But saving eight minutes in loading one of my own programs went a long way toward making a believer out of me.

The compact form used by the TC-8 in recording data on tape makes it possible to store 50,000 characters (bytes) on a ten-minute cassette, or 300,000 characters on a 60-minute cassette.

When you use the TC-8, you don't need to add dead space between the end of one program and the beginning of the next to permit you to position your tape by listening to the tone created by the data flow. Just position your tape anywhere ahead of the program you want to load and the TC-8 will find it—using the program's name you have assigned—and transfer it from tape to computer memory.

Furthermore, the TC-8 is practically immune to recorder volume setting problems. Any volume level setting between two and eight works just fine on my CTR-41.



TC-8 to recorder audio and control cable.

PARTS LIST

QTY	PART	RADIO SHACK NO.
2	1/8 in. MINIATURE PHONE PLUG	274-286
1	1/16 in. SUBMINIATURE PHONE PLUG	274-289
1	DIN, 5 PIN AUDIO PLUG	274-003
20 ft.	MICROPHONE CABLE WITH BRAIDED SHIELD	278-1277

The TC-8 Hardware

The Heath Company has been telling us for years that anyone who can follow simple directions can assemble their electronics kits. The people at JPC Products apparently feel the same way. They have so much confidence in their manual and kit that they guarantee to make the interface system work within 60 days of purchase if the buyer returns it to their plant in Albuquerque, NM. The only charge to the buyer is the cost of mailing the unit.

The TC-8 kit's assembly instructions are superb. JPC Products' instructions are clearer than those that accompany Heathkits. If you've built kits engineered by Heath Company, you will recognize that I've just paid JPC Products the ultimate compliment.

The manual contains a parts list, a picture of the printed circuit board with parts installed, a schematic diagram, many parts layout sketches, a short course in soldering techniques and a set of step-by-step parts assembly directions.

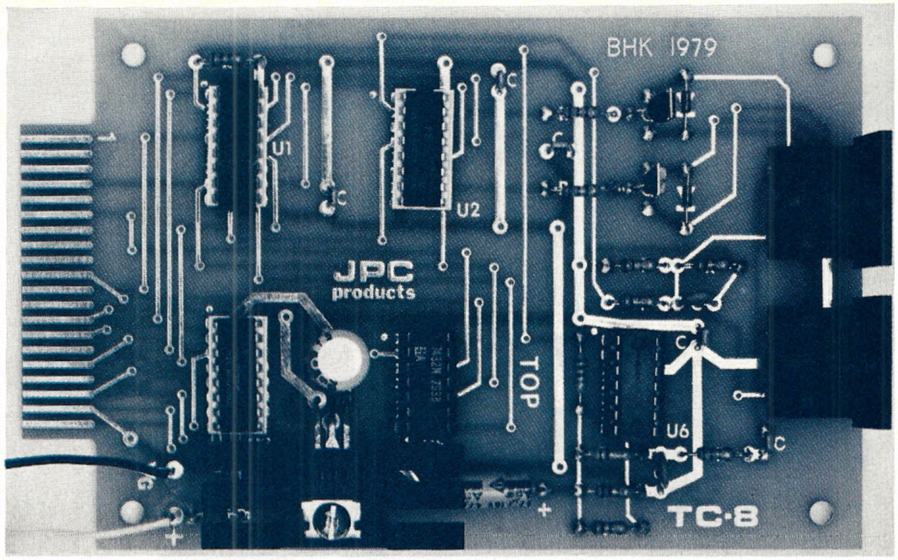
Assembly instructions (as well as the parts list) refer to resistors by their color

codes and ohmic values. Mounting and soldering instructions contain notes cautioning against using the wrong resistor.

The kit consists of a high-quality, double-sided, component-labeled circuit board, five integrated circuits (sockets provided), three diodes, two transistors, one voltage regulator, assorted resistors, capacitors, connectors, ribbon cable (connectors attached) and a power cord adapter (an encased step-down transformer).

The circuit board is mounted in a metal cabinet that measures 5-3/4 x 4-1/8 x 2-5/8 inches and is attached through a ribbon cable to the 40-pin connector located beneath the hinged door at the left rear edge of the TRS-80 Model I keyboard cabinet.

No modifications need be made to the TRS-80 to use the TC-8, so you needn't worry about voiding your Radio Shack warranty.



Five socketed ICs and other components mount easily on the TC-8's marked printed circuit board.

You can connect two cassette recorders to the TC-8 for use in recording or playback operations. I keep my CTR-41 recorder permanently connected to drive 1, using the audio and control cable assembly I built (see Fig. 1).

JPC Products' estimate of one hour to complete the assembly job—particularly for the neophyte builder—may be a bit overoptimistic. It took me almost two hours to assemble the kit. But I worked on the project in several short bursts, which is not the most efficient way to do it. I also scraped every resistor, capacitor, diode and transistor lead to remove any possible oxide buildup before I solder, so that takes additional time (but pays dividends in good connections).

When I asked Gerry Williams, president of JPC Products, whether the instruction manual could actually teach inexperienced kit builders to solder properly, he said that of more than 300 cassette interface kits sold thus far, only five had been returned for repair; only two of those five had developed problems that were traced to poor soldering.

The assembled kit system has worked perfectly since the moment I first turned it on. All components are of fixed value so there are no adjustments to be made.

The manual also provides clear instructions for modifying your CTR-41 recorder, if you have one, so that the TC-8 can control the recorder's tape drive motor. The modification simply involves interchanging two color-coded wires.

The TC-8 comes complete with its own 5-volt power supply, so it places no additional load on the TRS-80's power supply system.

The TC-8 Software

The TC-8 won't do a thing for you without the utility program—appropriately named "UTIL"—that accompanies it.

When you turn on your TRS-80, you answer the MEMORY SIZE? question by typing "31400" to reserve and protect space in the upper portion of computer memory for use by the 1354 byte UTIL program.

Because UTIL is a machine-language (binary) program, you must type "SYSTEM" and respond to the "*?" prompt by typing "UTIL" and by pressing the enter key. UTIL loads from the recorder connected to your TRS-80 in about 23 seconds. You then type "/" and press the enter key again to obtain the "UTIL READY" message.

Within the UTIL program is a shorter utility program named BOOT that occupies about 600 bytes of memory space. BOOT lets the TC-8 load a BASIC program from cassette tape if that program will need some—but not all—of the memory space that UTIL requires.

The TC-8 manual is unusually clear in describing how you use UTIL. Each step is explained and the computer's response—as viewed on the screen of your TRS-80 video monitor screen—is shown.

UTIL provides the following commands: SAVE, LOAD, LOAD?, LOADN, KILL, RSET, RUN, PUT, GET, GET?, GETN. In addition, UTIL supports rapid access sequential file management with these statements: OPEN, CLOSE, PRINT# and INPUT#.

The SAVE command is similar to CSAVE and is used to record BASIC programs. You must use a filename (up to eight characters in length), and you may specify which of two recorders connected to the TC-8 is to be used to record the program you are transferring from computer memory to cassette tape.

LOAD and LOAD? are similar to their CLOAD and CLOAD? counterparts except that they activate the tape recorder connected to the TC-8. As I mentioned earlier, when you use LOAD with your program's filename, you can position your cassette tape anywhere ahead of the program's location and the TC-8 will find and load it into memory.

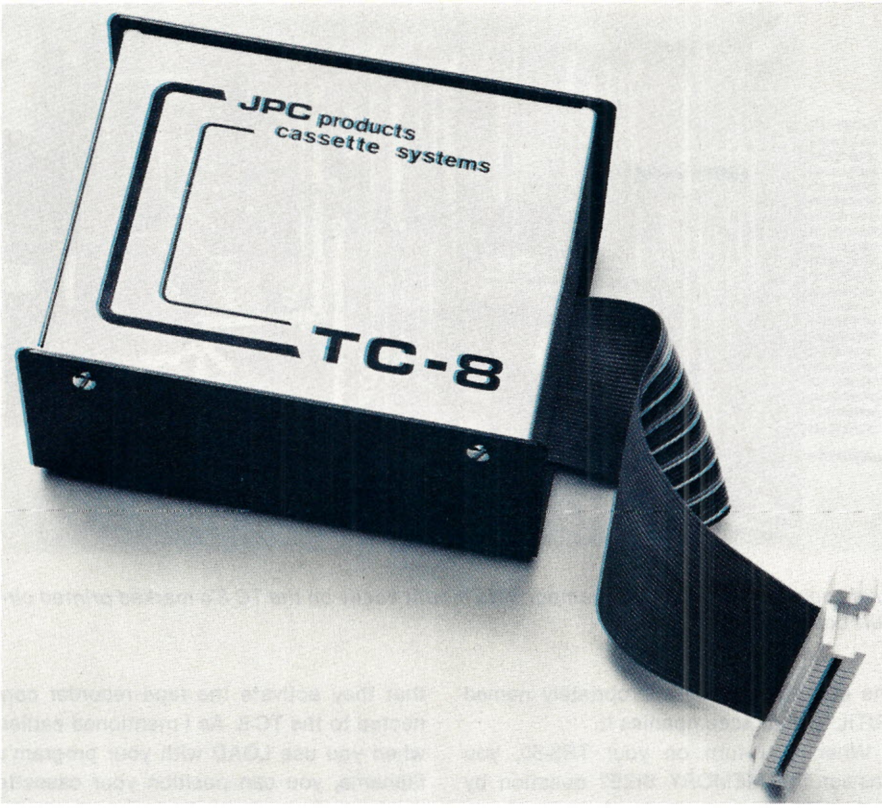
LOADN is a directory command that prints on your monitor's screen a catalog of all program names encountered while reading a cassette tape via the TC-8. Symbols appear beside each program filename shown to denote whether the program has been written in BASIC, machine language or source language.

LOADN is useful for positioning the cassette tape at the end of the last program so that another program can be recorded (SAVEd). This tape positioning is done by typing "LOADN" and the filename (in quotes) of the final program that currently resides on the cassette tape. The recorder's motor will stop at the end of the designated program.

The PUT command is similar to SAVE except that PUT is used to record machine-language programs via the TC-8. GET, GET? and GETN commands are similar to their LOAD counterparts and are used exclusively with machine-language programs.

If your BASIC or machine-language program will occupy all available space in your computer's memory bank, you may use the KILL command to free the space that UTIL occupies after you have used UTIL to load the program. After you use the KILL command, memory size is restored to 32767 bytes (for a computer having a 16K memory capacity).

An RSET command turns on your recorder so that its rewind and fast forward controls can be used to reposition cassette tape without removing the motor control



The TC-8 cabinet measures approximately 3 × 4 × 5 inches. Ribbon cable with 40 pin connectors attached is supplied.

plug. Depressing the TRS-80's break key terminates the RSET command and removes power from the recorder's motor. Owners of the CTR-80 recorder have no need for the RSET command.

The RUN command searches a tape, loads and executes a designated BASIC program via the TC-8 drive 1. Using RUN precludes your having to type "RUN" after your program has been loaded into memory.

The OPEN, CLOSE, PRINT# and INPUT# statements create and read sequential tape files at high (3000 baud) speed. You must specify the TC-8's cassette recorder drive number (1 or 2) when you use the PRINT# or INPUT# statements. String and numerical data may be intermixed in the file.

UTIL contains its own set of indicators, which appear in the upper-right corner of your video monitor's screen. Whenever you issue a command to UTIL, two dash marks appear. While UTIL is loading a program into computer memory, two asterisks appear; the rightmost one blinks rapidly.

If the program has been loaded successfully, the right-hand asterisk is replaced by an up-arrow symbol. If, for some reason, the program loads incorrectly, the blinking asterisk is replaced by C to indicate a checksum error, by M for memory error or by S for syntax error.

One little extra that reflects favorably on JPC Products' sensitivity to the needs of many of us TRS-80 owners is the inclusion in the TC-8 manual of a short glossary of computer terms. Words that are used to explain how UTIL commands and statements are employed are defined to enhance the user's understanding.

Nothing's Perfect

Although the TC-8's hardware works beautifully, the software program has one minor flaw.

Two pages of the instruction manual are devoted to helping you prepare a backup tape for the UTIL program. The intent is that the backup tape containing UTIL be used and that the master copy supplied with the TC-8 be stored in a nice, safe place.

The instructions for preparing the backup program cover using Radio Shack's Z-80 monitor program (TBUG) to combine the UTIL program with Radio Shack's keyboard debounce program (KBFIX). It's a great idea. Unfortunately, I couldn't get the combination UTIL/KBFIX program to respond properly to one command that the original UTIL program handles quite well. That command is KILL.

I am certain that JPC Products intended the backup program to support all of the commands that the master UTIL program provides. Many of us have programs that require so much memory space that they will not function after being loaded from the TC-8 until the KILL command has been activated to release the memory space occupied by UTIL.

But JPC Products has eliminated that bug. I have tested the revised UTIL program following the new instruction sheet, and the KILL command works perfectly.

Incidentally, the new version of UTIL has the keyboard debounce program built into it so there is no need to incorporate KBFIX yourself. No more bouncing keys while you are using UTIL.

I also had some difficulties when I used UTIL's PUT command. PUT requires that you specify in hexadecimal form the beginning address, the ending address and the transfer (execution) address of the machine-language program you want to save on tape using the cassette recorder you have attached to the TC-8. How many of us TRS-80 users know enough about machine language to be able to find those addresses for TBUG (monitor), EDTASM (editor/assembler), KBFIX (keyboard debounce) or other "SYSTEM" programs we may have purchased? I venture a guess: not many.

Again, JPC Products has recognized the problem and will supply with its UTIL program tape a monitor program named TINY that, among other things, will identify the machine-language addresses you will need to use the PUT command to save SYSTEM programs in fast TC-8 format.

TINY will also let those of you who have more than 16K of memory relocate UTIL at the high end of your 32K or 48K memory banks. TINY also provides a capability to examine and change data in memory that will make it unnecessary to load TBUG to perform those functions.

In the section of the manual that describes statements supported by UTIL, a sample program demonstrates how PRINT# (TC-8 drive) transfers a sequential file to cassette tape by using the OPEN, CLOSE and PRINT# statements. Another sample program shows how the INPUT# (TC-8 drive) statement reads data stored on tape. The two sample programs are unrelated. It would be much more instructive to those who are unfamiliar with sequential file creation and use if the data written to tape by the sample PRINT# program could be read back by the sample INPUT# program.

Conclusions

The TRS-80 Model I owner who still uses cassette tapes for program storage is sure to find the TC-8 recorder interface unit to be a real bargain at $93.50 (kit, plus shipping charges) or at $123.50 (assembled unit, plus shipping). The interface unit and the software that controls it are everything the manufacturer's advertisements have claimed—and more.

I never want to go back to loading my TRS-80 at 500 baud—not as long as my TC-8 will load it at least five times that fast.■

Frank J. Derfler Jr.
PO Box 691
Herndon, VA 22070

The Source
Revisited

Take a journey to the data communications capital and visit with The Source chairman of the board, Jack Taub.

Jack Taub, tightly wired dynamo and chairman of the board of STC.

Last month I discussed The Source in my article "What Is the Utility of a Utility?" (October 1980, p. 72). I recently visited the source of The Source—chairman of the board Jack Taub.

The Source claims to be something unique. They say they are a utility that pumps out information for everyone else's use, just like the other utilities pump out water or electricity.

I like to describe them as an interactive electronic newspaper. They give any user who has a terminal, modem and telephone such services as the news from United Press International; political analysis; business news and comment; guides for shopping, food, travel and restaurants; classified ads; and personal electronic mail. They provide this using the unique capability of the computer to quickly search, categorize and sort large amounts of data. Each user gets only the information requested exactly as he wants to receive it.

Users pay an initial fee of $100 and are charged a fee of either $15/hr. (prime time: 7 AM to 6 PM) or $4.25/hr. (non-prime time).

A Visit

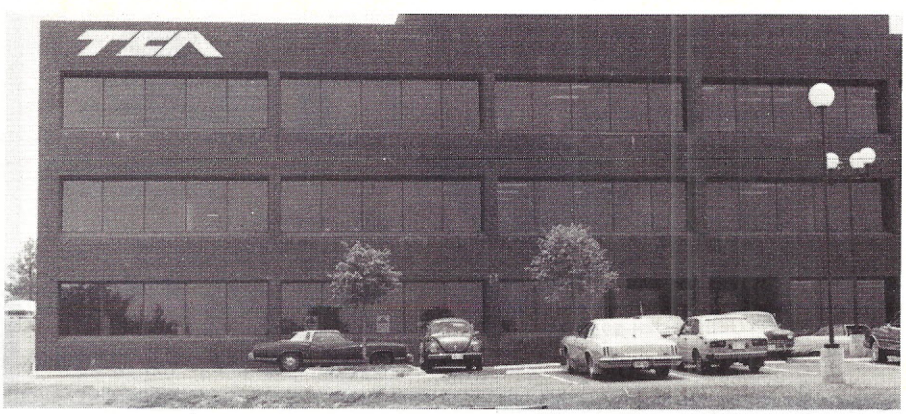
The offices of the Source Telecomputing Corporation are just outside Washington, D.C., in McLean, VA, snuggled up against the greatest giants of computing and communicating. This area is rich in telecommunications and equipment. Silicon Valley may be the U.S. center for computer technology, but northern Virginia has the corner on data communications. The Source is a natural product of this environment.

The staff of The Source is not large. I pictured rows of programmers and scores of corporate types bustling around. Instead I found an outfit with comfortable, but not plush, offices and little overhead.

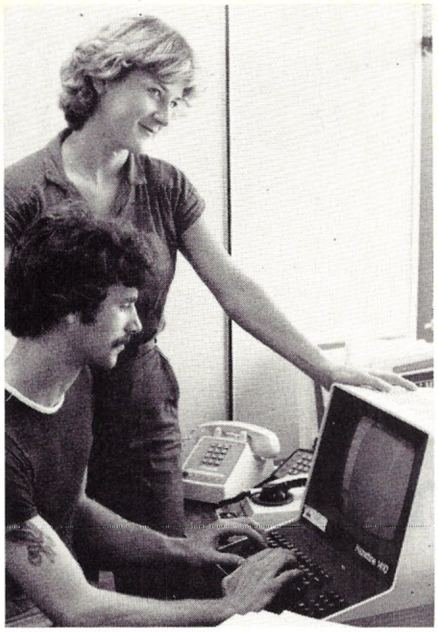
The Source's Prime computers are many miles away in Maryland, where they are maintained by a contractor. As I was to learn, this firm is not hardware-oriented, and the suggestion that my readers might like to see a picture of the hardware was dismissed by one executive because "It looks like a row of refrigerators." Later, I learned why the folks at the headquarters were thinking much more in terms of service and consumer impact than hardware.

Jack Taub is a dynamic individual I would describe as "tightly wired." He goes quickly from one subject to another and can resume a conversation in mid-sentence when interrupted. I was slightly hostile when we began because I had recently been the victim of a Source system crash, which caused the loss of at least an hour's work. But after a short time with Jack Taub, the system crashes didn't seem quite as important.

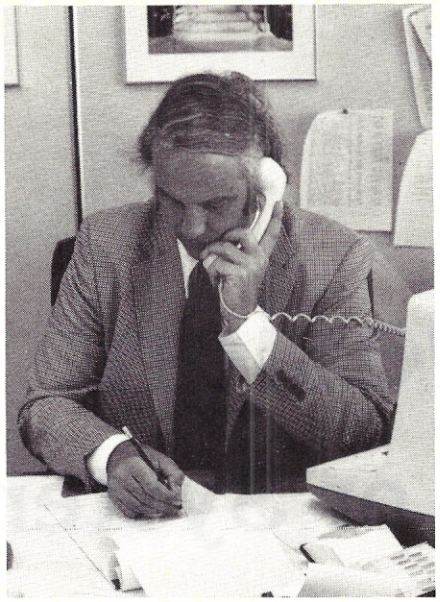
Taub: "We know that the transmission capability of the United States cannot support a full-scale information utility. Tymnet and Telenet cover the major cities, but a great portion of the country is without good access to our service. We are building our own network and will serve our customers in a variety of ways. We will soon be into the libraries of 1400–1500 communities in the U.S. We have signed an agreement with the Ohio College Library Consortium, which



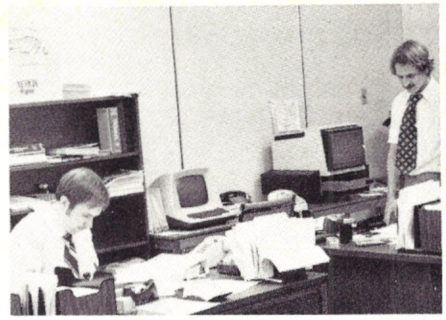
This building is the home of The Source corporate offices. The name changed from TCA to STC (Source Telecomputing Corporation) earlier this year. They are located in the heart of data communications country in McLean, VA.



This is the customer service branch of The Source. If you send a message to TCA088, either Avice Drumheller or Steve Salopek will try to help you with your problem.



Noel Jan Tyl is the corporate voice of STC. He publishes Sourceworld *magazine and edits all of the copy on The Source. He also wrote the new and much improved* Source Users' Manual and Master Index.



This small accounting office does all of the billing for Source services. The Source is billed to individual customers on their national credit card accounts. They seem to be saying, "Now what did we do with Derfler's bill?"

will provide many new communities with Source power."

Microcomputing: "But you can't provide reliable service to the customers you have now. How can you talk about 1500 new customers when the old ones suffer through system crashes?"

Taub: "Every minute of downtime is a minute too much for me. I worry about it, I'm sorry about it, but there are better things coming. We are in the same place the phone companies were in the 1920s. We are learning."

Microcomputing: "How about documentation and billing? Your documentation has been criticized, and your billing would give an accountant fits. The charges all come on one line with no breakdown."

Taub: "That's all true in part, but we have had very few complaints. People have been very kind to us. Every mistake we make in developing this information utility, every pain, every aggravation makes it better. We have no book to go by; we are writing the book. Some parts of the user's manuals become out of date almost as fast as we print them because the system is changing so fast. We have started a monthly magazine, *Sourceworld,* to try and keep our users up to date.

"Also, please consider this: Where else can you get even the computer utility power of The Source for the price you are paying? I could ask you, what do you expect for $4.25 an hour? But better things are coming."

Microcomputing: "OK, what is coming?"

Taub: "We have always intended to grow. You can't run a full-scale information utility by keeping everything on your own computers. We have recently signed a working agreement with Tymshare for a development and pilot operation, working toward establishment of The Source as a system practically unconstrained by the problems that affect it now. We will have greatly increased capacity for simultaneous users and great improvement in response times at all time. We will have much more capability, speed and redundancy."

Microcomputing: "So The Source will be a distributed system with the common data base chasing around the Tymshare system?"

Taub: "Yes, via the Tymnet network."

Microcomputing: "Will Telenet still be a Source carrier?"

Taub: "Of course, and we will still keep our local computers."

Microcomputing: "That is a whole new concept. The software will be tough to do."

Taub: "We use a lot of other people's software. That is why we have some inconsistency in our program commands and statements."

Microcomputing: "Yes, I never know if I should use 'Stop' or 'Quit' to end a program."

Taub: "But we are learning how to get commonality of commands even when we don't originate the software. We are also working on an on-line tutorial package which will take you by the hand and lead you through the features of the system."

Microcomputing: "That will be very valuable. How many users do you have now?"

Taub: "We have about 5000 paying customers. The majority of them operate in the off-prime-time periods."

Microcomputing: "There have been persistent rumors about huge financial losses and the possibility that you were in financial trouble. Would you like to comment on that?"

Taub: "The first management team went through a lot of money in a short time, but this is a new area and it is expensive to break ground. But, believe me, we are well-backed and the future is bright."

Microcomputing: "What is the future, beyond the Tymshare project?"

Taub: "You know, we could go out and grab all the smart people in this building and guess about the future all day and never come close. We can't see very far into the future because of what I call the begetting principle. One new idea begets another and that begets another and so on. We really can't even guess at what the forks in the road will lead to, but I do know The Source is going to be a tremendous vehicle for change. What all this change will eventually beget is certainly not clear to us now."

The people at The Source obviously see their business as much more than computer hardware or a computer service. They are learning and making mistakes, and there will probably be confusing and frustrating times ahead for both staff members and users. But The Source is doing things never done before, and those who stick through the experience will probably look back with fondness on their part in the history of data communications.■

Author's note: Shortly after this interview, STC announced Source 2, a system accessed through Tymnet. As of publication deadline, Source 2 had fast response time, but a much smaller data base than the original Source service.

6809 Design: Controller or System?

This chip is versatile enough for almost any application—from a simple black box controller to a complete disk-based business system.

Tim Ahrens
7405 Ladybug St.
Austin, TX 78744

Microprocessors have been traditionally broken up into two distinct groups: controllers and small personal systems.

Controllers can do everything from turning on lights to making better blends of gasoline. Some even count the number of French fries that go into each bag. As you can see, the microprocessor controller has many diverse applications in everyday life.

There are three elements to every controller—memory (ROM or RAM), I/O and the MPU. The ROM/RAM can be whatever size is necessary, and I/O can be either serial, parallel or both. The MPU should be easy to use, both in hardware and software. One of the best choices is the MC6809, the most advanced eight-bit microprocessor available.

A small system is an expanded controller and is used in applications ranging from hobby computers to small-business computers. They are single-user computers that run programs written in languages such as Pascal or BASIC. In some situations, software generation is the main purpose and is done with editors, assemblers and compilers. There are many more diverse controller applications than small systems, due to the nature of their environment.

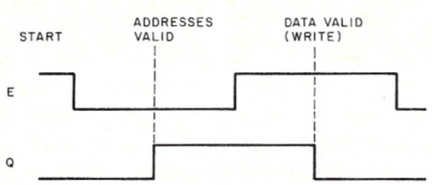


Fig. 1. E-Q relationships.

The MC6809—Hardware

The hardware features of the MC6809 make system design a snap. In traditional M6800 style, all peripherals are spoken to in a memory-mapped I/O fashion.

The MC6809 requires no complex clock generation devices: only a parallel resonant crystal across the Xtal and Extal pins with a frequency four times that of the bus. If you want an external frequency source, the Extal input will accept a TTL level of four times the bus frequency. Be sure to ground the Xtal pin when operating in this mode.

The crystal frequency is internally divided by four and then output on the E pin. In addition, a quadrature clock, Q, leads E by 90 degrees. (See Fig. 1.)

The falling edge of E signifies both the beginning and end of a cycle. On a read or write cycle, addresses, R/W and MPU status signals are valid on the rising edge of Q. This edge may be used to latch data. On a read cycle, data must be valid on the bus before the falling edge of E, which is late in the cycle. See the MC6809 data sheet for specific bus timing. Latching addresses or data is not required when using M6800 series peripherals, but interfacing to other devices may require these edges for timing purposes.

The reset input on the MC6809 is a Schmitt trigger input, which has a higher threshold voltage than standard peripherals. Peripherals thus come out of reset before the processor, and a simple R/C circuit resets the entire system. During power-on, reset should be held low until the clock oscillator is fully operational (about 100 ms). After that time, you may reset by holding the RESET line low for a minimum of one bus clock cycle.

Addresses are valid with the rising edge of Q. When the MPU doesn't need the bus for data transfer, it will output address $FFFF, R/W = 1 and BS = 0. Because of this, no VMA signal is used on the MC6809. If you want a retrofit to the MC6800 system, the VMA line may be tied high. The drive capability of the address bus in one Schottky TTL load and 90 pF. This makes single board design without buffers a reality.

The data bus provides bidirectional data transfers between peripherals and the MPU. The drive capability is one Schottky TTL load and 130 pF at related bus speed.

The HALT line will suspend program execution following the completion of the present instruction. When halted, BA goes high, indicating the address buses are in a high impedance state. Fig. 2 describes a simple single instruction stepper for the MC6809.

The MC6809 has four states that can be decoded by using the bus available (BA) and bus status (BS) pins:

BA	BS	MPU State
0	0	Normal (running)
0	1	Interrupt Acknowledge
1	0	Sync Acknowledge
1	1	Halt/Bus Grant

BA indicates that the MOS buses have been made high impedance, but does not mean that the bus will be available for more

than one cycle. BS, when decoded with BA, represents the MPU state.

The DMA/BREQ input lets you suspend execution and acquire the MPU bus for other uses, such as DMA and dynamic memory refresh.

A low level on the MRDY input pin allows E to be stretched in one-quarter bus cycle increments. This is useful when you are interfacing slow RAM, ROMs or peripherals to the bus. The maximum stretch is 10 us, due to the dynamic properties of the MPU.

The NMI, FIRQ and IRQ interrupt input pins provide the designer with methods of interrupting normal MPU operations.

NMI is the non-maskable interrupt pin. This input cannot be inhibited by the program. NMI finds general use in power-down applications, software refresh of dynamic RAM and real-time interrupt structures.

FIRQ is a fast maskable interrupt in the sense that only the program counter and condition code register are pushed upon the stack. The IRQ is an interrupt that can also be inhibited by program commands but will place all registers upon the stack when executed. For interrupt vector locations, see Fig. 3.

Software

While the MC6809 has hardware attributes, software is its forte.

The MC6809 gives you the following registers:

- two eight-bit accumulators, which can be concatenated into a single 16-bit wide register;
- two 16-bit indexable general-purpose registers;
- two 16-bit indexable-stack-type registers;
- one eight-bit direct page register; and
- one eight-bit condition code register.

See Fig. 4 for the MC6809 programming model.

Converting from 6800 to 6809 software is done by running the source code through a 6809 assembler or cross-assembler. Although the object codes for the 6800 and 6809 are noticeably different in most areas, numerous op codes have remained the same.

The addressing modes for the MC6809 are upward-compatible with the MC6800. The old modes have been kept and new ones added.

Direct addressing had previously been only in the lower 256 bytes of the memory map. This mode has been expanded to put that page anywhere in memory through the use of the direct page register (DPR). This register may be loaded with any value that will be the page in memory used for direct addressing.

For example, if the DPR contains \$02, then any instruction that uses direct addressing will have the value of 02 put on the address bus as the most significant byte. Following a system reset, this register is cleared to be compatible with the MC6800.

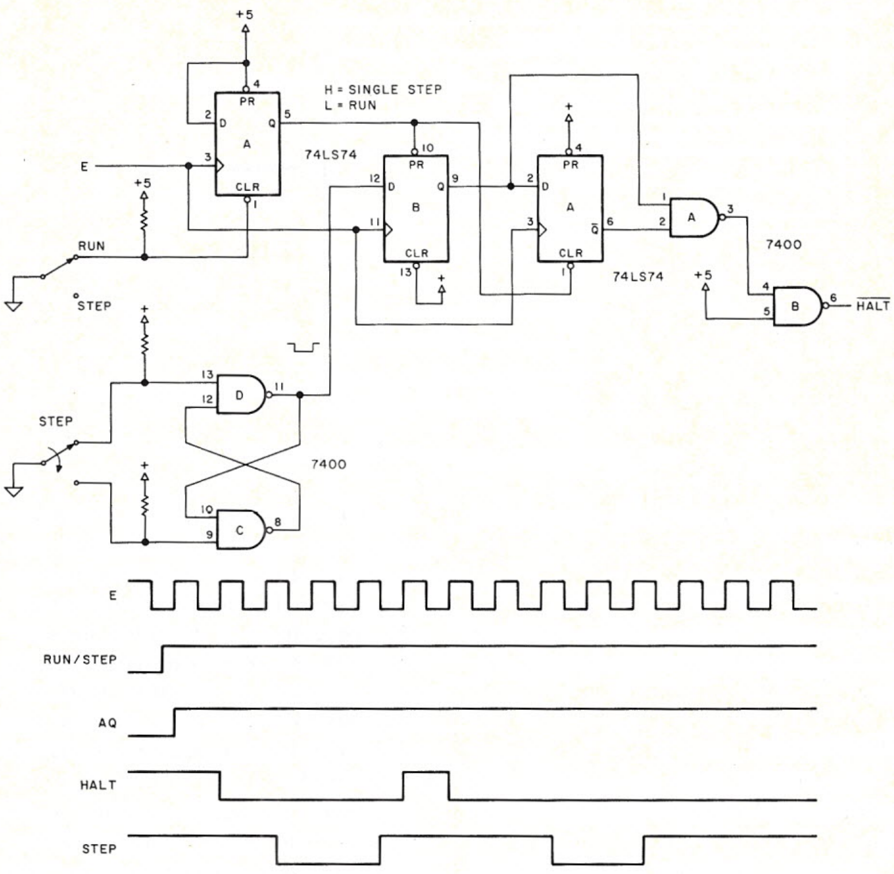


Fig. 2. Single stepper for the MC6809.

Relative branching had been limited on the MC6800 to -125 or + 127 bytes. In many cases, this restricted some programming applications and made position independent code (PIC) difficult without many alternate branches. The MC6809 allows relative branches to anywhere in the memory map (−32768 to +32767).

Another type of relative addressing is program counter relative. By using this mode, you can easily write position-independent code. For example, if you wanted to print a text string with the MC6800, the common method was:

```
LDX #MSG
JSR PRINT
```

Print is a routine within the code that prints text until you encounter an EOT character. This type of code is difficult to make position independent, but with the MC6809, PIC becomes very easy:

```
LEAX MSG,PCR
LBSR PRINT

MSG FCC/PRINT THIS/
```

The load effective address (LEAX) instruction takes the current offset from the program counter to the message, adds it to the PC and places it into the X register. Then, by doing a long branch to subroutine, the message gets printed. This code is fully position independent and thus executes properly anywhere within the memory map. The LEA instruction is available with any of the four indexable registers (X, Y, U or S).

The MC6809 has expanded index addressing modes, which include 0-, 5-, 8- and 16-bit constant offset, 8/16-bit accumulator offsets and auto increment/decrement. In addition, these indexing modes may have an extra level of indirection.

Indirect addressing is useful in many applications where addresses of parameters are taken on and off of the stack pointers. Such applications include higher-level languages such as Pascal and BASIC.

An example of how indirect addressing helps out when writing position independent code follows:

```
LDX#$E014  Loads X register with $E018 which is the address of the ACIA
PSHU X     Places $E018 on the U stack pointer
```

Now, any time data is to be loaded from

MS Byte	LS Byte	Function
FFFE	FFFF	RESET
FFFC	FFFD	NMI
FFFA	FFFB	SWI
FFF8	FFF9	IRQ
FFF6	FFF7	FIRQ
FFF4	FFF5	SWI2
FFF2	FFF3	SWI3
FFF0	FFF1	RESERVED

Fig. 3. Memory map for vector locations.

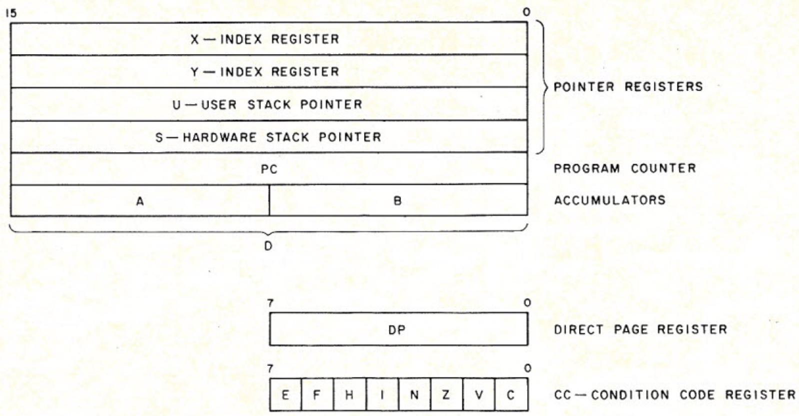


Fig. 4. Programming model of the MC6809.

the ACIA, only the following instruction is needed:

LDA [0,U]   Get data from ACIA

Note that many "addresses" may be placed on the stack and called in this indirect manner.

Miscellaneous

In the MC6809, any or all registers may be pushed onto the stack with a single instruction.

A multiply instruction multiplies the unsigned binary numbers in the A and B accumulator and places the unsigned result into the 16-bit D accumulator. This unsigned multiply also allows multiple-precision multiplications and takes only 11 machine cycles (5.5 us in a 2 MHz system).

The Basic Controller Design

What is required for a controller?

As mentioned earlier, the minimum is a microprocessor, program storage and I/O. The basic controller in this article contains an MC6809, two MC6821 parallel interface adapters (PIA), one MC6850 serial port (ACIA) and one EPROM of any desired density (MCM2708, 2716, 2532 or MCM68764). Also included is the necessary decoding and baud rate generation for the serial interface.

The bus frequency is 1.2288 MHz, corresponding to a cycle time of 813 ns. This frequency was chosen for one reason. 1.2288 × 4 = 4.9152 MHz, which is a common frequency and can be divided down by an MC14040 ripple counter to give most desired baud rates for the ACIA. Note that the bus speed is higher than that specified as the maximum rate for a standard MC6809. To be within specifications, an MC68A09 as well as A series peripherals are required.

To use standard 1 MHz parts, choose a 2.457 MHz crystal, which is still usable with the 14040. If the ACIA is not required or a different baud rate generation scheme is used, any crystal within frequency specifications may be used.

The decoding of this system is straightforward. If you anticipate no expansion over the original design, the 74LS42 may provide all necessary chip selects for the peripherals. The outputs of this 7442 are eight blocks of 8K. For a minimum parts count, tie each chip select of the RAM, ROM and peripherals to one of these outputs. The ROM must be the highest-order decode line ($E000 — FFFF).

Due to the incomplete decoding, each peripheral will occupy many locations within its respective block of memory. Here is an example of a decode scheme:

PIA 1	4000-5FFF
PIA 2	6000-7FFF
ACIA	8000-9FFF
RAM 1	A000-BFFF
RAM 2	C000-DFFF
ROM	E000-FFFF

Although incomplete decoding is used, it can be to your advantage. By addressing the lower bank of RAM in software as BC00 to BFFF and the upper bank of RAM as C000 to C3FF, you have 2K of contiguous RAM. This is possible due to the many mirror images that occur with incomplete decoding.

The R/W signal must be conditioned for use with 2114 RAMs. This conditioning effectively delays the valid R/W signal until the rising edge of E, which is halfway into the memory cycle.

The 74LS139 two- to four-line decoder is shown for those users who desire a more complete decoding scheme. By using the 139, these additional blocks may be decoded (see Fig. 5).

As mentioned earlier, the baud rates for the MC6850 are derived from the E clock through a CMOS counter. All common baud rates are available from 300 baud to 19.2 kilobaud, and if using the lower 2.45 MHz crystal, slide the taps down one for the correct baud rate.

RS-232 for the ACIA is provided by simple transistors, thus reducing cost over the traditional MC1488/89 receiver/transmitter devices.

A power-on reset circuit is provided in the 4.7k and 10 uF capacitor.

All unused inputs on the MC6809 are pulled up with 3.3k resistors for a wire-or capability. If you don't anticipate using these inputs, you can use a direct Vcc connection, further reducing cost and parts count.

You may further reduce the number of parts by using a device such as the MC6846, which includes 2K of mask programmed ROM, an eight-bit parallel I/O port and a 16-bit timer. Although this controller uses only a minimum of parts, its capabilities are great because of the flexible instruction set of the MC6809. See Fig. 6 for the complete schematic.

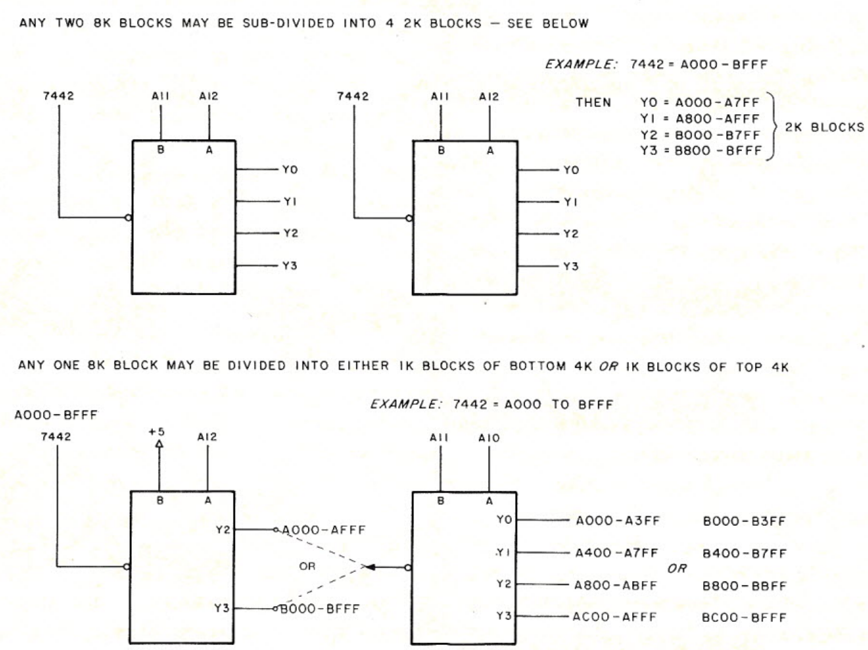


Fig. 5. 74LS139 additional decoding.

Load Your SWTP at 4800 + Baud

The author tried JPC Products' cassette interface and found it reliable to 9600 baud.

Jerry L. Hunt
6709 Forsythia
Springfield VA 22150

While your Kansas City Standard tape is loading, do you:

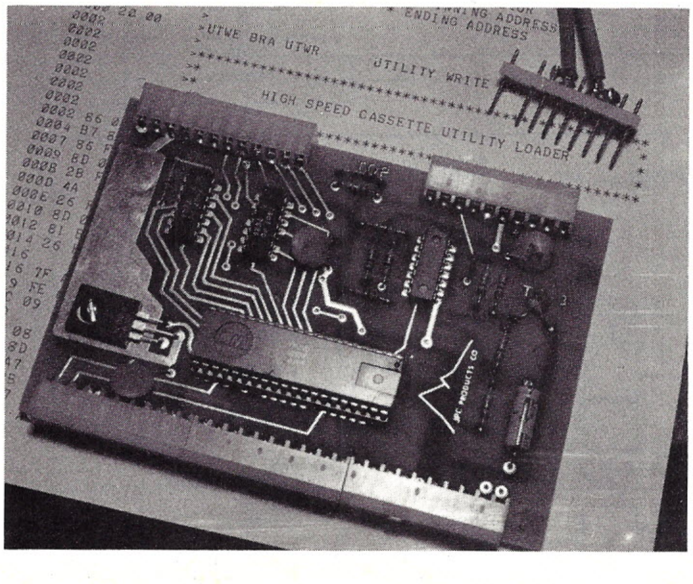
A. Tap your fingers impatiently?
B. Yell at your kids and dog?
C. Rebuild your keyboard?
D. Take a correspondence course in brain surgery?

If you would like to spend less time fussin' and fumin' and more time computin', read on.

Since I've had a computer, I've spent several man-days waiting for my KC tapes to load. This has become limiting, as well as irritating. After becoming fed up, I started looking for a medium with a bit more speed. My search first took me to the obvious devices such as digital tape decks and floppy disks. These gadgets have two common characteristics: quickness and expense. The first characteristic is very attractive, but the second is not as appealing.

One evening, while waiting for a tape to load and browsing through a *Microcomputing* magazine, I noticed an ad from JPC Products Co., PO Box 5615, Albuquerque NM 87185, for a $49.95, 4800 baud tape interface bit that plugged into an SWTP I/O port. I looked at the remaining 10 minutes of KC tape still to be loaded and ordered the interface!

About three weeks (and several more hours of KC tape loading) later, the package was delivered. It consisted of the hardware and a comprehensive hardware/software manual. The kit went together with ease. Hookup was equally easy and consisted of soldering two shielded cables to the connector and plugging them into a suitable cassette device.



TC-3 Hi-Speed Cassette Interface

Building Up Speed

Due to the high speed of the data flow—up to 9600 baud—two factors are important. High-quality tape is essential, as is a high-quality cassette machine. The manufacturer recommends only top of the line, low-noise tapes and provides a recommendation list of cassette recorders and decks. Basically, a good stereo tape deck and tapes should be used.

My way of providing these was to remove the stereo tape deck and tapes from my component stereo system. The deck has two features that are useful in this application: an accurate tape counter and vu meters (output meters). Also helpful were the record level and output level controls.

The software documentation provided included two programs: one for high-speed read and write and one for KC read. This type of interface is versatile as well as fast, since it functions almost entirely through software. Thus, it can be programmed for nearly any format, current or future! The data transfer rate is controlled by software constants and the computer's clock. A short program is included to determine your SWTP computer's clock rate, and constants are furnished so that the baud rate is variable up to 9600!

The manufacturer recommends the baud rate be set at 2400 for system setup, and once any bugs are exterminated, the rate is set to the advertised 4800 baud. After all the time I had sat listening to the whirring of my cassette recorder, this sounded like the speed of light!

Conclusion

I am immensely pleased with this system. I recommend it without reservation as the best buy in town for fast, economical off-line storage. My system cost me only $49.95 for the interface. If you need a good tape deck, add about $80 to that. So for less than $150 you can have a 4800 baud system capable of storing one megabyte (60-minute tape).

I have no association with JPC Products, except for admiring their product. I haven't even communicated with them, since the interface and software operate flawlessly.

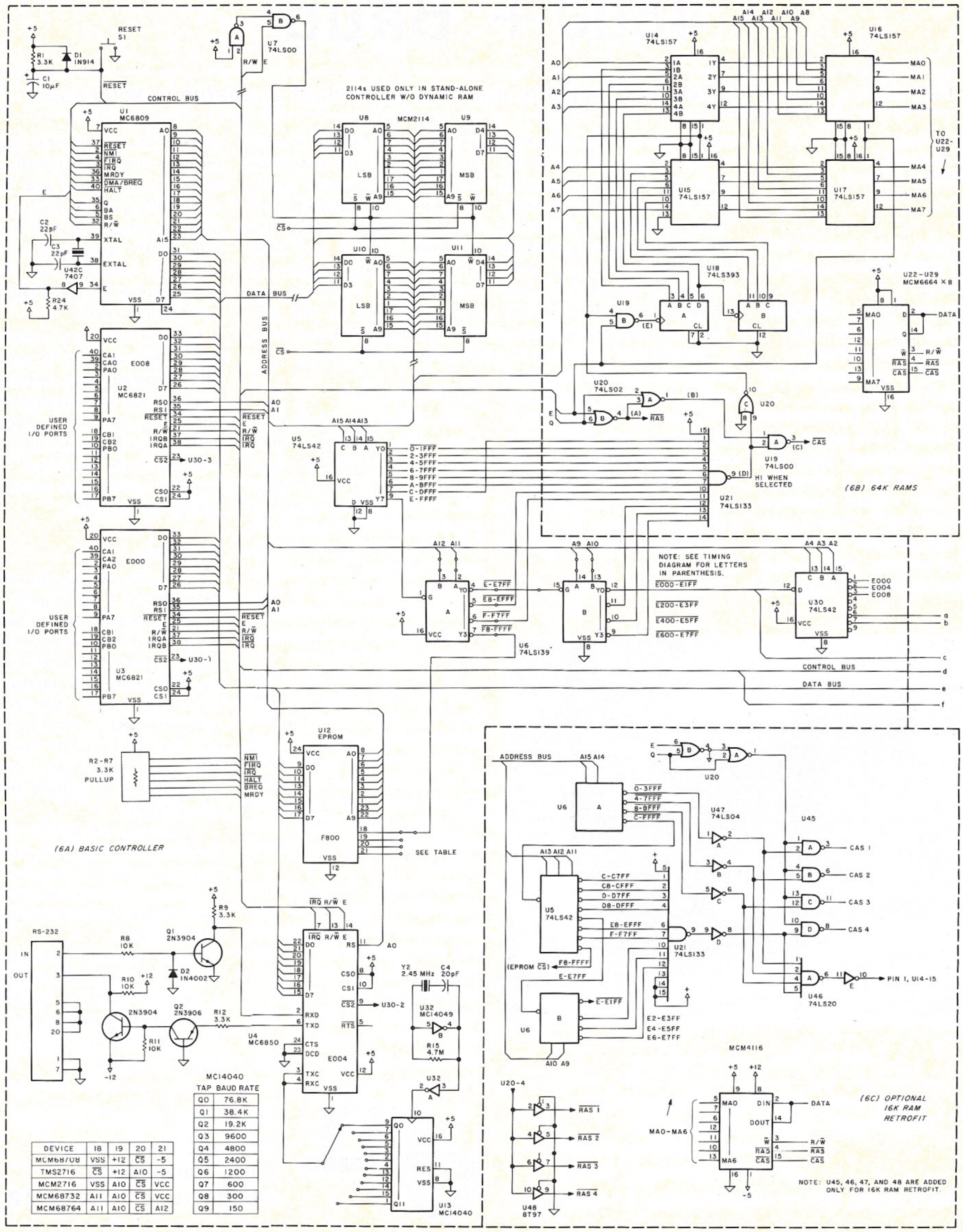
I have also just discovered that JPC is offering software for a cassette operating system, file handling and basic patches. My prayer is answered for about $27 on cassette! ■

Due to its functions, almost every system design goes through many changes or even a total redefinition of its intended use. The basic controller circuit described earlier can be easily launched into the small business/personal computer market with a few expansions. These include full 64K RAM enhancement and a floppy-disk controller for program storage/recall.

The RAM

The RAM expansion circuit uses the new MCM6664 64K X 1 dynamic RAMs, but the techniques employed may also be used with the more common MCM4116 16K X 1 dynamic chips. Also included is an easy

Fig. 6. Main schematic.



DEVICE	18	19	20	21
MCM68708	VSS	+12	$\overline{CS}$	-5
TMS2716	$\overline{CS}$	+12	A10	-5
MCM2716	VSS	A10	$\overline{CS}$	VCC
MCM68732	A11	A10	$\overline{CS}$	VCC
MCM68764	A11	A10	$\overline{CS}$	A12

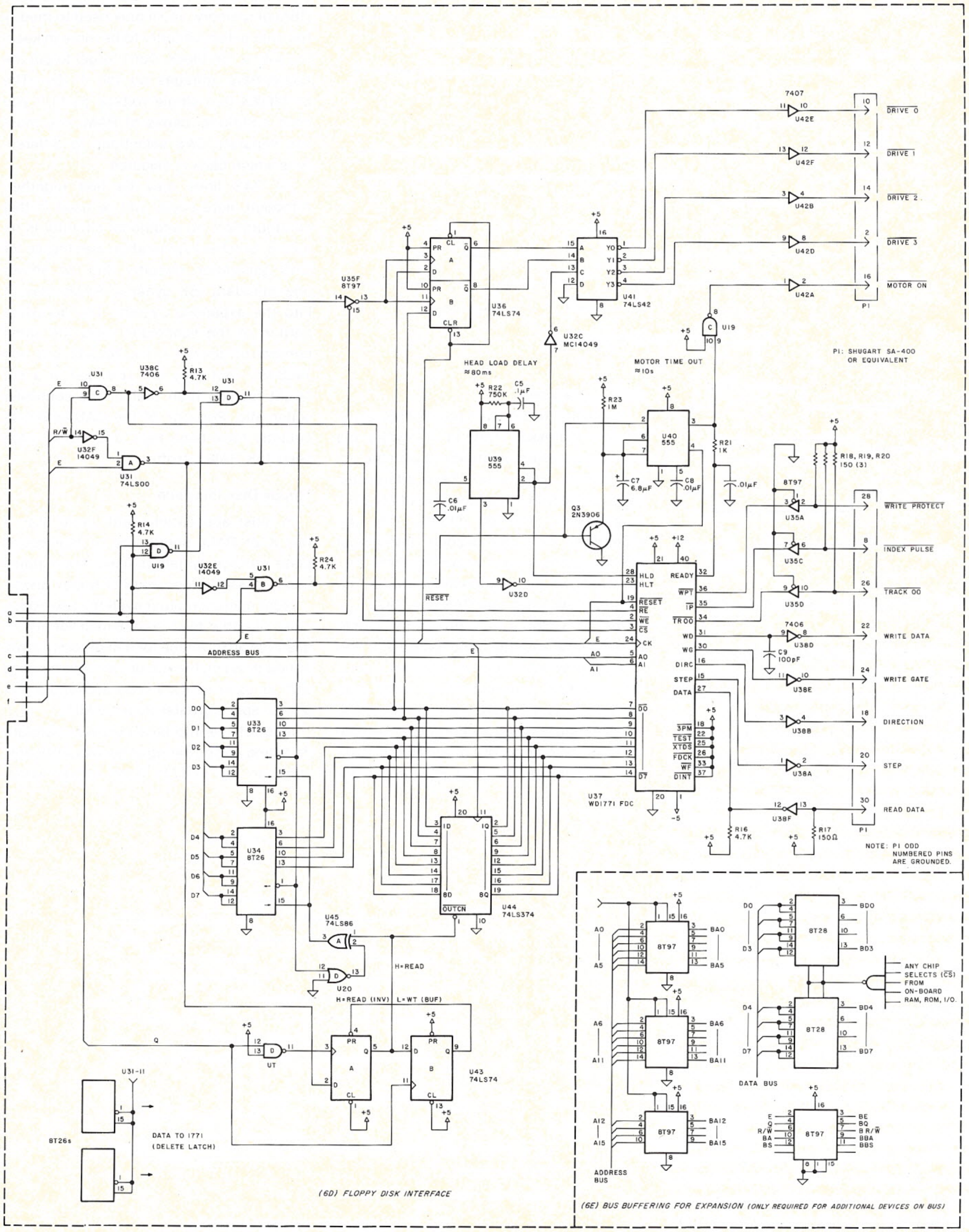
MC14040

TAP	BAUD RATE
Q0	76.8K
Q1	38.4K
Q2	19.2K
Q3	9600
Q4	4800
Q5	2400
Q6	1200
Q7	600
Q8	300
Q9	150

retrofit with 16K RAMs.

Dynamic RAMs, unlike their static counterparts, require a periodic "refreshing" to maintain integrity of the stored data. This refreshing can take on several different forms, one of which is discussed here—RAS only refresh. Fig. 7 shows how the RAS only refresh technique is used with the 6664s.

The dynamic RAMs have only eight address input lines, which select the desired memory cell within the chip. These address lines are multiplexed; that is, half of the addresses (the rows) are "strobed" in during the first part of the cycle, and the other half (the columns) are strobed in later in that same cycle. The waveforms in Fig. 8 show their relationship in the cycle.



(6D) FLOPPY DISK INTERFACE

(6E) BUS BUFFERING FOR EXPANSION (ONLY REQUIRED FOR ADDITIONAL DEVICES ON BUS)

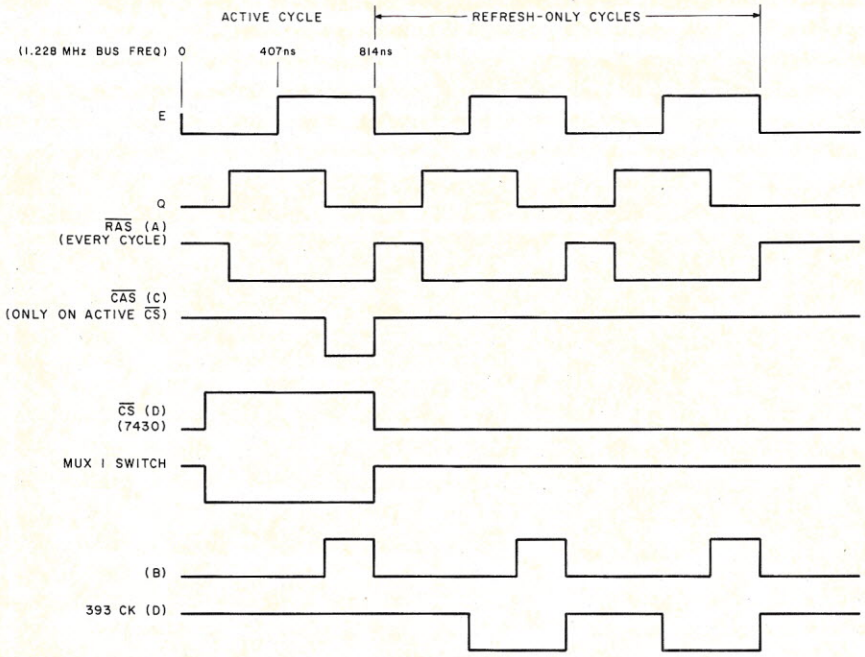


Fig. 7. RAS-only refresh timing.

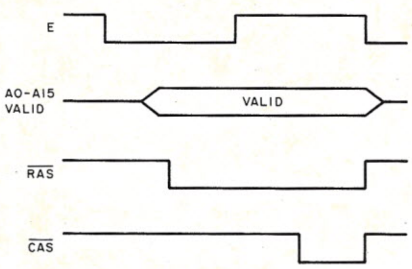


Fig. 8. RAS and CAS relationships.

As shown, the row addresses are strobed in on the falling edge of RAS, and the column addresses are strobed in on the falling edge of CAS. On a write cycle, the data should be valid on the falling edge of CAS, and on a read cycle, data comes out of the RAMs on the rising edge of CAS.

To retain the stored data, the RAMs must have every row accessed within 2 ms. Since a program execution generally does not access these rows within the required time period, you must use a hardware design to help out the refresh. The refresh schematic shows a pre-multiplexer, which selects either the regular row addresses from the MPU or a pseudo row address supplied by an external counter.

When the RAM is selected for a memory operation, the normal rows pass. At all other times, these pseudo addresses are supplied and are continually counting through the 128 rows. During the time that real addresses are being multiplexed, the clock signal going to the binary counter is held high until the pseudo addresses are required. Operation in this fashion ensures that all rows are accessed in an increasing manner, and no rows will be passed over during an access.

Following the decision point of normal or pseudo addresses (early in the cycle), the row addresses are multiplexed with the upper column addresses. These addresses, which come from the second set of 74LS157 multiplexers, are fed directly into the MCM6664s, which are decoded into actual memory cell locations within the RAM. Fig. 7 also shows the relationship between the multiplex switches and RAS and CAS.

The CAS signal is supplied by a chip select signal and the combination of E and Q. A chip select signal is obtained from a 13-wide NAND gate. The inputs to this gate come from appropriate address decoders. This CAS signal controls the actual data going into and coming out of the RAMs. Data must be valid on the falling edge of CAS (for a write), and data is valid on the rising edge of CAS (for a read). See Fig. 6b for the entire 64K RAM schematic.

16K RAMs

The design used for the 64K RAMs can also be applied to standard 16K X 1 dynamic RAMs. If you need only one bank (16K) of memory, you'll only need to modify the chip select circuitry to be more in keeping with a 16K block. Don't forget to put the appropriate voltages on the 4116s. (The MCM6664 is a single voltage part.) If you need additional banks of RAM, you must use separate CAS selections to differentiate which bank is selected.

All RAS lines may be tied together. Although more power will be used in this configuration, no additional circuitry is required for refresh generation. See Fig. 6c for CAS generation circuitry. Fig. 6c shows how standard 16K dynamic RAMs may fit into the expanded system. The decoding portion of the schematic uses the same number of devices—one 74LS42 and one 74LS139—but they are arranged in a different fashion than that of the controller schematic. Portions of the CAS selection circuitry have been kept, and others have been 16K RAM retrofit.

Floppy Disk Interface

In most applications with more than a few K of RAM, some type of high-speed mass storage system is used. Many times this is cassette tape, hard or floppy disks.

Most microcomputer systems use floppy disks of either the 5¼- or 8-inch variety. I'll describe an interface for a minifloppy drive, although an eight-inch drive could be used with an external data separator and a processor speed greater than 1.5 MHz.

Most of the interface involves standard decoding and buffering of necessary buses, although the FD1771 does require some

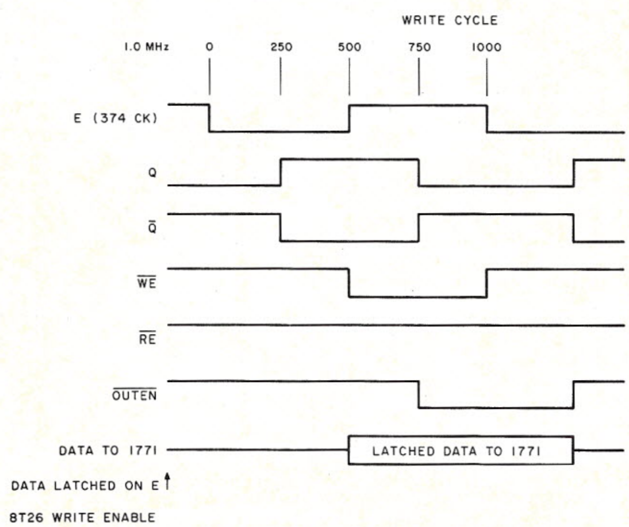


Fig. 9. Floppy disk controller timing.

strange circuitry to work with the MC6800 or MC6809.

The first is that of the R/W line. The 1771 uses separate read and write enable signals. These are derived from the R/W line and E. Each of these signals is valid for the entire E high time.

The other circuit is required for latching data into the FDC on a write cycle. The 1771 data sheet states that data must stay valid on the data bus for at least 150 ns after the WE pulse goes high. This data is valid for only 30 ns on the MC6809, or about 10–20 ns on the MC6800/6802. Because of this, a latch is needed to hold the data on the bus. In the read mode, no circuitry is required because the 1771 holds the data for more than the minimum that is specified by the MC6809.

This data-hold specification of the MC6809 denotes how long the bus drivers of the MPU are actually turned on, and not how long the data will be held on the bus. This time on the bus can vary, mainly with the amount of loading that is present. If TTL or other heavy-load devices are present on the bus, the decay time will be shortened.

But, if all that are present are MOS parts and other high-impedance devices, this hold time will traditionally be until the MPU bus drivers are driven to their opposite states (as early as the next cycle). In the given example, U44, U43, U45A, U20D and U7D may be taken out (see Fig. 6d). To be within the guaranteed specifications of both the 1771 and MC6809, these parts must be installed, but, in this application, the characteristics of a MOS bus may be used to your advantage to save PC board space and parts count (see Fig. 9).

Fig. 6d shows the schematic used as the floppy disk interface. You can use standard Shugart SA-400 or equivalent disk drives.

Minifloppies generally use a dc motor for the diskette drive motor, thus shortening their useful operating life. To make more efficient use of this time and to save the oxide on the diskette, you can turn off the drive motor when not accessed.

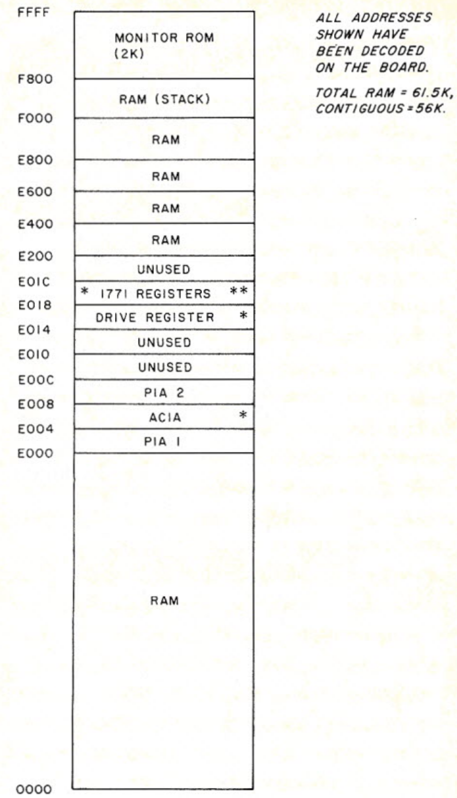
U40 (MC1455) turns on the drive motor when location $E018 is accessed. This is the base location of the 1771, so any access on the FDC will restart the drive motor. During any nonactive 1771 time, the 555 will hold the drive motors on for about ten seconds before shutting down. This time is determined by the value of C6.

Another 1455 provides the head load timing delay. This time is about 80 ms, which gives the head enough time to settle before signaling the 1771 that data transactions may take place.

Drive selection is determined by U36 and U41. U36 provides a way to latch information from the data bus. This information is the drive number and is sent to U41, which

```
                              *  THIS 'MINI-MONITOR' IS ALL THAT IS REQUIRED
                              *  IN THE EXPANDED SYSTEM TO PROVIDE THE USER
                              *  WITH A BOOT FOR THE 'FLEX' DISK OPERATING SYSTEM.
                              *
                              *  THE SYSTEM AUTOMATICALLY BOOTS UP FROM RESET TO
                              *  THIS ROUTINE.  OTHER ROUTINES MAY BE PUT INTO
                              *  ROM, PROVIDING THE USER WITH MORE CAPABILITIES.
                              *
                              *  THIS CODE IS COURTESY OF TSC INC.
                              *
                        E014  DRVREG  EQU   $E014     DRIVE REGISTER
                        E018  COMREG  EQU   $E018     COMMAND REGISTER OF 1771
                        E01A  SECREG  EQU   $E01A     SECTOR REGISTER OF 1771
                        E01B  DATREG  EQU   $E01B     DATA REGISTER OF 1771
                              *
F800                                  ORG   $F800
                              *
F800 B6     E018              START   LDA   COMREG    TURN MOTOR ON
F803 86     00                        LDA   #0
F805 B7     E014                      STA   DRVREG
F808 8E     0000                      LDX   #0000
F80B 3D                       OVR     MUL             DELAY FOR SPEED UP
F80C 30     1F                        LEAX  -1,X
F80E 26     FB                        BNE   OVR
F810 C6     0F                        LDB   #$0F      RESTORE
F812 F7     E018                      STB   COMREG
F815 8D     2B                        BSR   RETURN
F817 F6     E018              LOOP1   LDB   COMREG
F81A C5     01                        BITB  #1
F81C 26     F9                        BNE   LOOP1
F81E 86     01                        LDA   #1
F820 B7     E01A                      STA   SECREG
F823 8D     1D                        BSR   RETURN
F825 C6     8C                        LDB   #$8C      READ WITH LOAD
F827 F7     E018                      STB   COMREG
F82A 8D     16                        BSR   RETURN
F82C 8E     C000                      LDX   #$C000
F82F C5     02                LOOP2   BITB  #2        DRQ?
F831 27     05                        BEQ   LOOP3
F833 B6     E01B                      LDA   DATREG
F836 A7     80                        STA   0,X+
F838 F6     E018              LOOP3   LDB   COMREG
F83B C5     01                        BITB  #1        BUSY?
F83D 26     F0                        BNE   LOOP2
F83F 7E     C000                      JMP   $C000
F842 8D     00                RETURN  BSR   RTN
F844 39                       RTN     RTS
                              *  RESTART VECTORS
FFFE                                  ORG   $FFFE
FFFE F800                             FDB   START
                                      END

0 ERROR(S) DETECTED

SYMBOL TABLE:

COMREG E018    DATREG E01B    DRVREG E014    LOOP1  F817    LOOP2  F82F
LOOP3  F838    OVR    F80B    RETURN F842    RTN    F844    SECREG E01A
START  F800
```

Mini-monitor listing.



*COMPATIBLE WITH TECH. SYST. CONSULT. 6809 FLEX DOS.

**1771 REGISTERS:

	READ	WRITE
E018	STATUS	COMMAND
E019	TRACK	TRACK
E01A	SECTOR	SECTOR
E01B	DATA	DATA

Fig. 10. System memory map.

decodes which drive is to be selected.

System Thoughts

The system timing signal E is used by all peripherals, including the 1771 for data transfers. For a controller or other small system, clock rates of up to 2 MHz may be used with the MC6809. Note, however, that the 1771 will not work much above 1.25 MHz when used in the shown configuration. I am using a 4 MHz crystal on the MC6809 and 2.45 MHz crystal for baud rate generation. The memory map for the entire expanded system is shown in Fig. 10.

Expansion

Although RAM expansion for this system over 64K is not practical, except with address translation circuits, other devices such as EPROM programmers, I/O cards, graphics cards and printer driver cards may be necessary in an expanded or small-business system configuration. Fig. 6e shows how the address/data buses may be buffered to supply the necessary signals for other cards on the bus. This bus may be anything that is close at hand, or it may be the standard Exorciser or SS-50 bus. Note that no buffering of the address or data buses is required on the single board expanded system because of the drive capabilities of the MC6809. With no software, the most elaborate piece of hardware is reduced to a pile of junk.

Rather than write an entire disk operating system (DOS), which might take me forever, I looked into the systems already available for the MC6809. Flex from Technical Systems Consultants proved to be the best choice as a DOS from both a cost and capabilities viewpoint. TSC has consistently featured excellent software at an affordable price ever since the advent of the MC6800. The new 6809 Flex has kept all of the capabilities of the standard 6800 Flex, so a conversion from an existing MC6800 system would not be too great. TSC also offers a wide range of Flex-compatible software, which includes an extended BASIC and an extensive debug package.

Since almost all operations use Flex, a small monitor ROM is all that is required. Any software debugging operations may be done with the debug package. The monitor ROM contains the following functions:

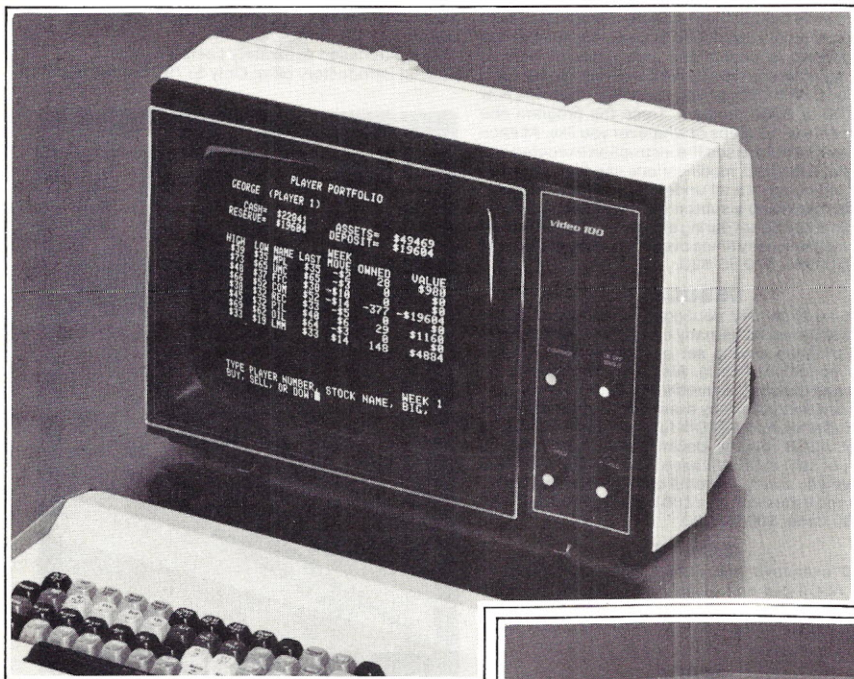
INITIALIZE FLEX

Now, that's a small monitor program! The monitor may be put in almost any type of ROM but must be placed at the top of memory so the MC6809 may get the appropriate restart vectors. See the monitor listing.

Conclusion

In these days and times, it doesn't take much to make a complete system. Whether 64K or 16K RAMs are used, this design can fill many requirements of either the controller or small systems market. ■

PROFESSIONAL B/W MONITORS
Designed for industry...priced for the home.



video 100

The video 100 computer monitors are ideal for all your personal and business needs. These highly reliable 12" black and white monitors feature a 12 MHz band width and 80 character by 24 line display. Plug-in compatability with Apple, Atari, Radio Shack, O.S.I., Micro-Term and Exidy make these the perfect text display for almost any system.

Sturdy, lightweight plastic cabinet

UNDER $170.00

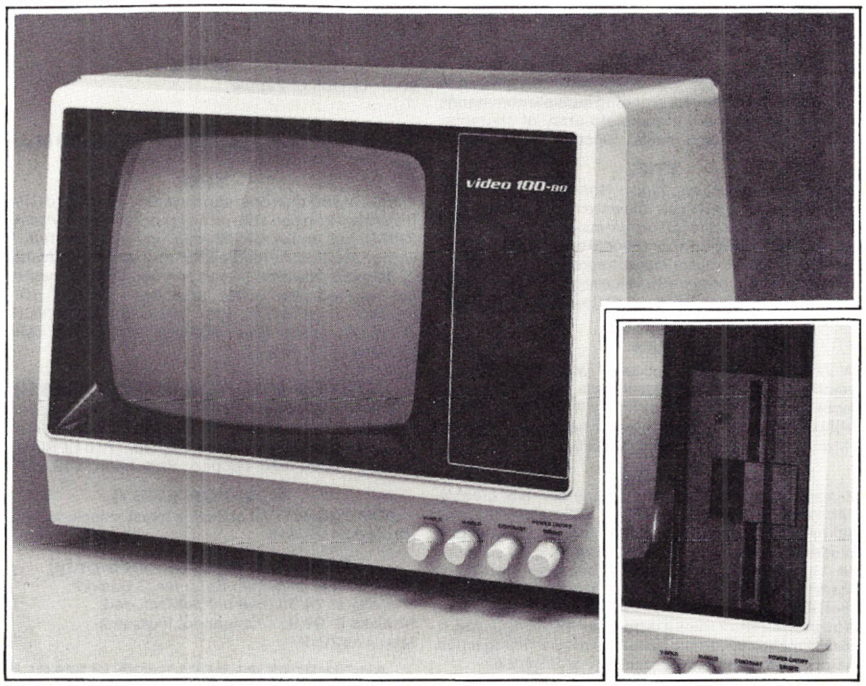
video 100-80

The model 80 features an industrial grade metal cabinet with built-in disk mounting capability and space for an 11" x 14" PC board for custom designed electronics.

The solid state circuitry assures a sharp, stable, and trouble-free picture. The front panel controls include power, contrast, horizontal hold, vertical hold, and brightness. Adjustments for size, video level, and width are located on the rear panel.

Rugged metal cabinet with disk space

UNDER $200.00



VIDEO 100 AND VIDEO 100-80 SPECIFICATIONS

- 12" diagonal measure display
- Convenient front panel controls
- Video bandwidth 12MHz ±3 DB
- Input impedance 75 Ohms
- 80 character by 24 line display
- 90° deflection picture tube
- Video 100-80 provides mounting space for mini floppy disk.
- Resolution—Over 700 lines at center horizontally—over 350 lines at center vertically

LEEDEX CORPORATION ✓59

2420 East Oakton • Suite E • Arlington Heights, Ill. 60005 (312) 364-1180 • TLX: 25-4786

Dealer discount available

All About ASCII

With data communications networks bringing the world within the grasp of microcomputers, a fuller understanding of the ASCII character set becomes increasingly important.

Thomas W. Parsons
42 Willow Place
Brooklyn, NY 11201

The world of the microcomputer user is exploding. From an 8080 in a box, we have gone to high-level languages, floppy disks and operating systems. We have an S-100 standard and a growing variety of compatible processors, memories and peripherals, with inexpensive megabyte storage on Winchesters the latest arrival.

The next development appears to be the computer network, and with this, your system will no longer be confined to your home or office. With a modem and a telephone, you will be able to tie into the whole world—read wire-service dispatches, interrogate data bases (financial, scientific, medical), communicate by electronic mail, make your own travel reservations.

Many people will be content to sit at a terminal and use these services. But to get the most out of this world and to tie your computer into these networks, you will want to know the details of how these systems work.

Talking to a network with your computer is a small part of data communications. This is a big field, and the first step is to understand the language that your computer must use to talk to a data network. This language is the ASCII character set, particularly the control characters that form what we might call the invisible part of the ASCII code.

ASCII stands for "American Standard Code for Information Interchange." In almost any book on computers, you will sooner or later come across the information in Tables 1 and 2. Many people know why this code exists and how it works, but few know much about the control characters. What do DLE, SYN or GS mean, and what are they for?

Until recently, it didn't much matter to the micro world, because the significance of control-C, for example, was a private matter between the user and, say, CP/M. ASCII, however, is basically a communications code, and now that data communications is beginning to reach out to the small user, these codes are going to be more than just casually interesting.

When you start to look into the ASCII code, many other questions crop up. For example, why are the codes seven bits long when eight bits is such a natural size for a byte? Why do the characters appear in the order they do, and how were they selected? All of these questions have answers, but it takes a little digging to find them.

I did most of my digging in the standards that define the code. These are published by the American National Standards Institute (ANSI), the Consultative Committee on International Telephone and Telegraph (CCITT) and the International Organization for Standardization (ISO). With all due respect to these organizations, these stan-

	First hexadecimal digit							
Second hexadecimal digit	0	1	2	3	4	5	6	7
0	NUL	DLE	SP	0	@	P	`	p
1	SOH	DC1	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3	#	3	C	S	c	s
4	EOT	DC4	$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(	8	H	X	h	x
9	HT	EM	)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[	k	{
C	FF	FS	,	<	L	\	l	|
D	CR	GS	-	=	M	]	m	}
E	SO	RS	.	>	N	^	N	~
F	SI	US	/	?	O	_	o	DEL

Code 27: Apostrophe or acute accent
2C: Comma
2D: Hyphen
5F: Underline
60: Grave accent

Table 1. Table of ASCII character codes in standard format.

DEC	OCT	HEX	NAME	KEY*	DEC	OCT	HEX	KEY	DEC	OCT	HEX	KEY
0	0	0	NUL	^1	43	53	2B	+	86	126	56	V
1	1	1	SOH	^A	44	54	2C	,	87	127	57	W
2	2	2	STX	^B	45	55	2D	–	88	130	58	X
3	3	3	ETX	^C	46	56	2E	.	89	131	59	Y
4	4	4	EOT	^D	47	57	2F	/	90	132	5A	Z
5	5	5	ENQ	^E	48	60	30	0	91	133	5B	[
6	6	6	ACK	^F	49	61	31	1	92	134	5C	\
7	7	7	BEL	^G	50	62	32	2	93	135	5D	]
8	10	8	BS	^H/BS	51	63	33	3	94	136	5E	^
9	11	9	HT	^I/TAB	52	64	34	4	95	137	5F	_
10	12	A	LF	^J/LF	53	65	35	5	96	140	60	`
11	13	B	VT	^K	54	66	36	6	97	141	61	a
12	14	C	FF	^L	55	67	37	7	98	142	62	b
13	15	D	CR	^M/CR	56	70	38	8	99	143	63	c
14	16	E	SO	^N	57	71	39	9	100	144	64	d
15	17	F	SI	^O	58	72	3A	:	101	145	65	e
16	20	10	DLE	^P	59	73	3B	;	102	146	66	f
17	21	11	DC1	^Q	60	72	3C	<	103	147	67	g
18	22	12	DC2	^R	61	73	3D	=	104	150	68	h
19	23	13	DC3	^S	62	74	3E	>	105	151	69	i
20	24	14	DC4	^T	63	77	3F	?	106	152	6A	j
21	25	15	NAK	^U	64	100	40	@	107	153	6B	k
22	26	16	SYN	^V	65	101	41	A	108	154	6C	l
23	27	17	ETB	^W	66	102	42	B	109	155	6D	m
24	30	18	CAN	^X	67	103	43	C	110	156	6E	n
25	31	19	EM	^Y	68	104	44	D	111	157	6F	o
26	32	1A	SUB	^Z	69	105	45	E	112	160	70	p
27	33	1B	ESC	ESC	70	106	46	F	113	161	71	q
28	34	1C	FS	^\	71	107	47	G	114	162	72	r
29	35	1D	GS	^`	72	110	48	H	115	163	73	s
30	36	1E	RS	^=	73	111	49	I	116	164	74	t
31	37	1F	US	^–	74	112	4A	J	117	165	75	u
32	40	20	SP	SPACE	75	113	4B	K	118	166	76	v
33	41	21		!	76	114	4C	L	119	167	77	w
34	42	22		"	77	115	4D	M	120	170	78	x
35	43	23		#	78	116	4E	N	121	171	79	y
36	44	24		$	79	117	4F	O	122	172	7A	z
37	45	25		%	80	120	50	P	123	173	7B	{
38	46	26		&	81	121	51	Q	124	174	7C	|
39	47	27		'	82	122	52	R	125	175	7D	}
40	50	28		(	83	123	53	S	126	176	7E	~
41	51	29		)	84	124	54	T	127	177	7F	DEL
42	52	2A		*	85	125	55	U				

*Diablo 1640 keyboard. The character ^ indicates use of the control key.

Table 2. Table of ASCII character codes with alternate number systems and keystrokes for control characters.

dards must rank as some of the least thrilling reading in the world. I am going to try to summarize them, leaving (I hope) the boring parts behind. The standards, along with a couple of more readable books, are listed in the references at the end of this article.

The ASCII Character Set

Tables of the ASCII code come in two shapes. Table 1 is used in the standards and shows the structure of the code more clearly. I prefer Table 2, because it isn't tied to any one number system and because it gives the keystrokes used for generating the nonprinting characters.

Otherwise, the two tables are basically the same. For example, the line feed (LF) is encoded with the bit pattern, 0001010. If this pattern is interpreted as a binary number, it has the value 10, and opposite 10 in the decimal column of Table 2 you will see LF. The number 10 in hexadecimal is 0A, and in column 0 and row A of Table 1 you will also find LF. The keystroke combination assigned to LF is control-J (written ↑J in the table), and if you will try this on any standard terminal, you will see that it works.

Set the terminal to "local" so you don't have to be connected to the computer, hold down the control key and strike J; you will see that the terminal advances to the next line. It is a bother to use a control-J every time you want a line feed, so most terminals supply a special key for this function; in such cases Table 2 gives two alternate keystrokes for the character.

Table 1 is organized into eight columns, and the control characters are all grouped together in the first two columns. The remaining columns contain graphic characters: the letters, numbers and punctuation marks that we ordinarily think of as being the whole point of the ASCII code. (Graphic simply means printable here and does not necessarily refer to the drawing of pictures.)

The control characters provide all of the auxiliary information that goes with any message transmitted to some remote station. People could simply add comments such as "this is the beginning of the next message" or "start a new page here." But it is more economical to implement these comments as special symbols and you don't have to strip them out of the text later on.

Types of Control Characters

We can group the control characters into several different types. One type controls the layout of the text on the printed page. These format controls include the backspace and the horizontal and vertical tabs. (For some reason, the space is considered a nonprinting graphic character rather than a format control.)

Another type of control, less well understood than the formatters, manages the transmission of data. These codes most clearly show ASCII's basic function as a communications code. They include an inquiry code, a yes and a no code and a number of symbols for marking the different parts of a message. You can attach a header to each message, giving, for example, the addressee's name and location, and the transmission controls provide ways of identifying the header and marking where it ends. Other controls mark other structural divisions within the message.

Similar to the transmission controls are the information separators, intended to mark logical subdivisions within a text. Then there are miscellaneous codes, such as the one that rings the bell or the one that marks the end of medium (similar to the EOF or tape mark used on magnetic tape) or the device controls used for turning devices on and off.

Four especially interesting controls are ESC, DLE, SO and SI. These allow two kinds of extension to the ASCII set.

The ESC character announces that the following codes form part of an escape sequence. The codes in an escape sequence do not have their normal meanings; instead, the sequence as a whole has its own special meaning. ESC sequences are frequently used to control equipment.

For example, CRT terminals with addressable cursors use ESC sequences to control the cursor location. Daisywheel printers use ESC sequences to set tab stops, margins and other options.

Most device manufacturers seem to invent their own ESC sequences as they need them, but there is a move to standardize how ESC sequences shall be formed and used; you can find information on this in

ANSI X3.41-1974 (see the references for the titles of these standards). For controlling transmission facilities, another whole family of sequences begins with DLE (for data link escape). These sequences are thoroughly standardized, since they are used on communication channels that serve many different users; details on DLE sequences are given in ANSI X3.28-1976.

The SO and SI characters allow an even more sweeping extension. SO (shift out) announces a switch to a whole new code in which all the bit strings have some other set of meanings instead of their standard ones. These new meanings continue until an SI (shift in) appears, at which point the codes revert to their regular meanings. No one in my reading has specified what these new meanings will be, but only the graphic character set will be switched. This seems reasonable, since presumably any conceivable code would always require the controls in the first two columns. (In most Centronics printers, SO enables those impressive-looking double-width characters, and SI returns the printer to standard width again.)

In Table 3, I have provided a glossary of all of the control characters. This table mainly reflects their accepted meanings, but again you should remember that many private users have found their own uses for these codes.

To cite just two examples, a number of operating systems (CP/M, among others) use NAK (control-U) to cancel a command line and ETX (control-C) to halt execution of a program. You will see that the ASCII meanings of these control characters have little or nothing to do with these uses. But there is absolutely nothing wrong with this; a standard should be followed only as long as it serves the interests of those concerned. Nevertheless, in this article I am interested in explaining all these characters from the point of view of data communications.

You will notice one significant omission in Table 3. Every terminal has on its keyboard a break key, but there is no ASCII code for BREAK. Why not?

The reason is that break is not a character in the ordinary sense. It is a special signal, originally intended to interrupt the other party's transmission in case of emergency. When ASCII characters are transmitted over voice-grade telephone lines at low rates, the bit patterns are transmitted one bit at a time at some uniform rate. Generating a break bypasses this process and sends out one long pulse that might be thought of as a drawn-out zero. This pulse doesn't fit the normal bit pattern, and when it is detected at the other end, it is recognized as a break.

Designing the Code

How does a code like this get set up?

A standard is usually drawn up by a committee, composed of representatives of concerned bodies (manufacturers, users, universities) and other interested individuals, all of whom work without pay under the auspices of the standard-setting organization.

I served on one such committee a couple of years ago. Most of our work was done by mail. Our chairman started out by soliciting suggestions and comments on the existing version of the standard. About once a month we would receive a big, fat envelope full of photocopies of everyone's latest opinions and suggestions. We would read them through, attack or second others' suggestions and defend or concede our own. Our comments, mailed back to the chairman, then contributed to the next month's big, fat envelope. (None of my own suggestions survived.)

Finally, we came to a consensus of sorts, and a final report went out to the sponsoring body—in our case, the ACM.

I assume that the ASCII standard was drawn up similarly, although perhaps their envelopes were fatter, since there were many more interested parties. Certainly they found many more serious problems to grapple with than we did, and it is interesting to consider what some of their problems were.

The ASCII code had to conform to a number of different requirements, not all of which were consistent. Some of the more important of these requirements were the following:

1. The code had to be as small as possi-

ACK (acknowledgement)—generally yes answer to various queries, but also sometimes means "I received your last transmission and I'm ready for your next."
BEL (bell)—causes bell, beeper or other audible alarm to sound.
BS (backspace)—moves carriage or cursor back one position.
CAN (cancel)—indicates that previous material is to be disregarded. (Specifically, how much material this refers to is a matter that must be decided on by the users.)
CR (carriage return)—moves carriage or cursor back to beginning of line.
DC1-DC4 (device controls)—for control of user's terminal or similar devices. No standard functions assigned, except that DC4 frequently means stop. (CCITT suggests a number of possible assignments; in general, they prefer using the first two controls for "on," the last two for "off," and DC2 and DC4 to refer to the more important device. In an earlier system, these codes were labeled X ON, TAPE, X OFF and $\overline{\text{TAPE}}$, respectively. X stood for "transmitter," and TAPE and $\overline{\text{TAPE}}$ stood for "tape on" and "tape off." These labels are still found on the keytops of some terminals.)
DEL (delete)—used to delete a character. (Called RUB OUT on some terminals. Not strictly speaking a control character, since it does not appear in column 0 or 1 of the ASCII table.) Assignment of this to the all-ones bit pattern is historic: the only way to erase a bit pattern punched into paper tape was to punch out all the holes and agree that the resulting pattern was equivalent to a null. ASCII still considers DEL equivalent to a null, although many operating systems use it to erase the preceding character.
DLE (data link escape)—introduces a special type of escape sequence specifically for controlling the data line and transmission facilities.
EM (end of medium)—means that this is the end of the paper tape (or other medium) or that this is the end of the material on the medium.
ENQ (enquiry)—usually used to request identification or status information. (In older systems, this code was sometimes called WRU—"Who are you?")
EOT (end of transmission)—marks the end of transmission after one or more messages.
ESC (escape)—marks the beginning of an escape sequence—a series of codes which as a group have a special meaning, usually a control function. (Called ALT MODE on some terminals.)
ETB (end of transmission block)—it may be convenient to break a long message up into blocks. ETB is used to mark block boundaries. (Usually the blocks have nothing to do with the format of the message being transmitted.)
ETX (end of text)—marks the end of a text. (See SOH.) Used to be called EOM, "end of message," and may be so labeled on some terminals.
FF (form feed)—advance to top of next page.
FS, GS, RS, US (file, group, record and unit separator)—a set of "information separators" provided for delimiting chunks of information. There is no standard usage imposed, except that FS is expected to refer to the largest division and US to the smallest.
HT (horizontal tab)—tabs device to next predetermined stop on the same line. (It's up to the users to decide where the horizontal and vertical tab stops are to be.)
LF (line feed)—moves carriage or cursor down one line. (Some systems combine carriage return with LF, and the combination is then called new line (NL).)
NAK (negative acknowledge)—means "no" answer to various queries; or sometimes, "I got your last transmission, but it had errors and I am awaiting retransmission."
NUL (null)—used mainly as a space filler. (See also SYN.)
SI (shift in)—used after SO to indicate that codes revert to normal ASCII meanings.
SO (shift out)—indicates that the bit patterns to follow will have meanings outside of the standard ASCII set and will continue to do so until SI is encountered.
SOH (start of heading)—it is assumed that any message will consist of a heading (stating the name and location of an addressee) and a text. SOH marks the beginning of the heading. Used to be called SOM, "start of message."
STX (start of text)—marker for beginning of text and end of heading (if any). Used to be called EOA, "end of address."
SUB (substitute)—character used to take the place of a character known to be wrong.
SYN (synchronous idle)—some high-speed data communication systems use synchronized clocks at transmitter and receiver. During idle periods, when there are no bit patterns to enable the receiver's clock to track the transmitter's, the receiver may drift out of sync. Every transmission following an idle period is therefore prefaced by three or four SYN characters. The SYN code has a bit pattern that enables the receiver not only to lock onto the transmitter's clock, but also to determine the beginning and end points of each character. SYN characters may also be used to fill short idle periods in order to maintain synchronization—hence the name.
VT (vertical tab)—tabs device vertically to next predetermined stop.

Table 3. Definitions of ASCII control characters.

ble while still accommodating all alphanumerics, control codes and a reasonable selection of punctuation. (This was the reason for using seven bits—an eight-bit length provides 256 codes, which they considered unnecessarily large. Some thought has been given to extending the code to eight bits, nevertheless.)

2. The code had to be extendable. This was accomplished by including the SO and SI codes, thus allowing the user to switch over to an alternate code, and by providing the ESC and DLE characters for encoding various control functions as escape sequences.

3. The alphanumerics had to be coded so that sorting the bit patterns as if they were binary numbers would automatically alphabetize the corresponding characters. (This alphabetization sequence is called the collating sequence.) This is more complicated than just having A come before B. For example, the blank has to come before everything else so that Roberts will alphabetize ahead of Robertson. Also, the comma should precede the alphanumerics so that Roberts, K. is ahead of Robertson. It is also standard practice to put the digits at the end of the collating sequence, but this was one of the considerations that proved incompatible with other requirements.

4. Special characters were to include a complete set of punctuation marks, all regular business and mathematical signs, all special characters used in the major programming languages (except APL) and a complete set of accents, or diacritical marks, for the principal European languages. They ran out of space, and the sacrifices came mostly from the last category. We are left with the tilde (õ), the circumflex (ô) and grave (ò) accents. The apostrophe was made to double for the acute (ó) accent, the comma for the cedilla (,) and the quotation marks for the umlaut, or diaeresis (ö). (CCITT recommends that these last three symbols be interpreted as diacriticals whenever they are preceded or followed by a backspace—as they would be if they were overprinted on some other character.)

5. It had to be possible to extract reasonable subsets from the ASCII code for special purposes by truncating it to six bits and also to get a suitable arithmetic subset by truncating to four bits. In the latter case, for example, by mapping the codes from hex 2A to hex 39 onto the number 0 through 15, you get the digits plus the decimal point and a complete set of arithmetic operators, missing only the equals sign.

6. There were a number of miscellaneous considerations, of which I will list only two or three examples here. ACK and NAK were located far apart so a "yes" answer was not likely to be turned into a "no" by transmission errors. A space is the most common sort of information separator, so the information separation codes were located so that the lowest-order one would be next to the space.

Special symbols were paired where possible to match the pairing of symbols on a standard typewriter keyboard (for example, / and ?, which usually appear on the same key). Special characters were distributed so that when this pairing broke down, it did so on infrequently used characters. (The pairing corresponds to a one-bit difference in the bit patterns and was intended to simplify keyboard design. With the advent of cheap ROMs and of powerful microelectronics generally, this is probably not as important a requirement as it used to be.)

ASCII and You

What is the importance of this to the average micro user? It's always useful to understand the workings of the system you use, and I know from experience that the first nontrivial character manipulating you try will land you right in Table 2, looking for the decimal equivalent to some nonprinting character (usually ESC).

But in addition, the ASCII set is literally the ABC of data communications. So far, communications for the average small user, going at 30 characters per second over regular phone lines, is pretty simple. But we can expect these systems to increase in speed and sophistication, and when they do, programming for them will undoubtedly use the ASCII controls for handshaking. ■

1. American National Standards Institute, *American National Standard Code for Information Interchange*, ANSI X3.4-1977; *American National Standard Code Extension Techniques for Use with the 7-Bit coded Character Set of American National Standard Code for Information Interchange*, ANSI X3.41-1974; *American National Standard Procedures for the Use of the Communication Control Characters of American National Standard Code for Information Interchange in Specified Data Communication Links*, ANSI X3.28-1976.

2. Consultative Committee on International Telephone and Telegraph (CCITT), Sixth Plenary Assembly, Vol. VIII.1, *Data Transmission over the Telephone Network*, Recommendation V.3, International alphabet #5.

3. International Organization for Standardization (ISO), *7-Bit Coded Character Set for Information Processing Interchange*, ISO 646-1973.

4. J. E. McNamara, *Technical Aspects of Data Communication*, Maynard, Mass., Digital Equipment Corporation, 1977.

5. J. Martin, *Systems Analysis for Data Transmission*, Englewood Cliffs, Prentice-Hall, 1972.

FMG CORPORATION

P.O. Box 16020
Fort Worth, Texas
76133
(817) 294-2510

✓70

VISA® master charge

FMG CORPORATION NOW CARRIES GRAHAM-DORIAN & PEACHTREE SOFTWARE

M-530

NEW VERSATILITY

For Your TRS-80

CP/M® 2

CONTROL PROGRAM FOR MICROCOMPUTERS ENABLING YOU TO RUN SOFTWARE PUBLISHED FOR CP/M 1.4 ON THE TRS-80

CP/M is considered the industry standard disk operating system because it gives you the hardware-independent interface you need to make your computer work for you. CP/M 2.0 is the latest in the evolution of a proven reliable and efficient software system. FMG CORPORATION NOW OFFERS THE CP/M 2.0 FOR THE TRS-80. It features an enhanced upward compatible file system and powerful new random access capabilities. The CP/M 2.0 from FMG provides the ability to run software published for the CP/M system, on the TRS-80 Model II. From minidisks, floppy disks, all the way to high-capacity hard disks, the flexibility of CP/M 2.0 makes it a truly universal operating system. The package includes an 8" system disk, editor, assembler and debugger for the TRS-80

Available in Format A, B, C, G only . . . $200/$25

MP/M™

MULTI-PROGRAMMING MONITOR

NEW INDUSTRY STANDARD

A deluxe operating system that provides big computer facilities at small computer prices. MP/M is a monitor program which operates with your microcomputer to provide multi-terminal access with multi-programming at each terminal. Best of all, it's CP/M compatible which means you can run a wide variety or programming languages, applications packages, and development software.

You can run simultaneous editors, program translators, and background printer spoolers. Or you can use MP/M for data entry or data-base access from remote terminals. Or you can use MP/M real-time features to monitor an assembly line and automatically schedule programs for execution throughout the day. MP/M makes an excellent focal point for a cluster of connected microcomputers. The possibilities are limitless.

(Format B) . . . . $450/$35
(Format G) . . . . $300/$35

*CP/M and MP/M are trademarks of Digital Research.
Z80 is a trademark of Zilog, Inc.
TRS-80 is a trademark of Tandy Corp.
Pascal/M is a trademark of Sorcim.

All FMG Software Products Include All Necessary Manuals

• MICROPRO INTERNATIONAL

SOFTWARE WITH MANUAL / MANUAL ALONE

(M) **SUPER-SORT I** — Sort, merge, extract utility as absolute executable program or linkable module in Microsoft format. Sorts fixed or variable records with data in binary, BCD, Packed Decimal, EBCDIC, ASCII, floating & fixed point, exponential, field justified, etc. Even variable number of fields per record! . **$225/$25**

SUPER-SORT II — Above available as absolute program only . . . . **$175/$25**

SUPER-SORT III — As II without SELECT/EXCLUDE . . . . **$125/$25**

WORD-STAR — Menu driven visual word processing system for use with standard terminals. Text formatting performed on screen. Facilities for text paginate, page number, justify, center and underscore. User can print one document while simultaneously editing a second. Edit facilities include global search and replace, Read/Write to other text files, block move, etc. Requires CRT terminal with addressable cursor positioning . . . . **$495/$40**

WORD-STAR Customization Notes — For sophisticated users who do not have one of the many standard terminal or printer configurations in the distribution version of WORD-STAR . . . . **NA/$95**

WORD-MASTER Text Editor — In one mode has superset of CP/M's ED commands including global searching and replacing, forwards and backwards in file in video mode, provides full screen editor for users with serial addressable-cursor terminal . . . . . . **$150/$25**

FLOPPY SAVER — Protection for center holes of 5" and 8" floppy disks. Only 1 needed per diskette. Kit contains centering post, pressure tool and tough 7 mil mylar reinforcing rings for 25 diskettes.

5", Kit **$14.95**
5", Rings only **$7.95**
8", Kit **$16.95**
8", Rings only **$8.95**

HEAD CLEANING DISKETTE — Cleans the drive Read/Write head in 30 seconds. Diskette absorbs loose oxide particles, fingerprints, and other foreign particles that might hinder the performance of the drive head. Lasts at least 3 months with daily use.

8" **$32.00**
5¼" **$30.00**

(M) • **DESPOOL** — Allows flexibility and efficiency. (Disk file printing can be accomplished while simultaneously using the computer for other tasks) Slower printers do not tie up the computer. Requires 32K minimum . . . . . . **$75/$10**

(M) • **SCREEN EDIT** — Text editor for program entry — allows user the ability to see entries as they are being made. Has command which enables user to move the viewed position of the file anywhere within the current data file OR add information anywhere in the file. Requires 16K minimum . . . . **$125/$25**
(Also available in TRS DOS format.
Specify model or TRS-80)

(M) • **MAC** — Disk-based, powerful macro assembler utilizes Standard Intel Mnemonics. Includes macro processor.

The CP/M 8080 Macro Assembler reads assembly language statement from a diskette file and produces an Intel "HEX" format object file on the disk suitable for processing in the TRS-CP/M environment. Requires 32K minimum and CP/M . . . . **$100/$25**

(M) • **ZSID** — Efficient and reliable program testing system for Z80 microcomputers. Capabilities include traceback and histogram facilities. Allows real time break points.

ZSID is a symbolic debugger which expands upon the features of the TRS-CP/M standard debugger, providing greatly enhanced facilities for assembly language program check-out. Requires 32K minimum and CP/M . . . . . . **$99/$25**

(M) **MAIL LIST** — Mailing list maintenance package. No sorting required to print normal address labels in zip code sequence. Supports new larger zip code. Sorts and selects on multiple fields. Labels may be printed in user selectable formats. Includes sort and select utilities **$300/$35**

• PEACHTREE SOFTWARE SYSTEMS

(T) **GENERAL LEDGER** — Records details of all financial transactions. Generates a balance sheet and an income statement. Flexible and adaptable design for both small businesses and firms performing client writeup services. Produces reports as follows: Trial Balance, Transaction Registers, Balance Sheet, Prior Year Comparative Balance Sheet, Income Statement, Prior Year Comparative Income Statement and Department Income Statements. Interactive with other PEACHTREE accounting packages. Supplied in source code for Microsoft BASIC . . . . **$990/$30**

(T) **ACCOUNTS PAYABLE** — Tracks current and aged payables and incorporates a check writing feature. Maintains a complete vendor file with information on purchase orders and discount terms as well as active account status. Produces reports as follows: Open Voucher Report, Accounts Payable Aging Report and Cash Requirements. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft. BASIC . . . . **$990/$30**

(T) **ACCOUNTS RECEIVABLE** — Generates invoice register and complete monthly statements. Tracks current and aged receivables. Maintains customer file including credit information and account status. The current status of any customer account is instantly available. Produces reports as follows: Aged Accounts Receivable, Invoice Register, Payment and Adjustment Register and Customer Account Status Report. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC . . **$990/$30**

(T) **PAYROLL** — Prepares payroll for hourly, salaried and commissioned employees. Generates monthly, quarterly and annual returns. Prepares employee W-2's. Includes tables for federal withholding and FICA as well as withholding for all 50 states plus up to 20 cities from pre-computed or user generated tables. Will print checks, Payroll Register, Monthly Summary and Unemployment Tax Report. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC . . . . **$990/$30**

(T) **INVENTORY** — Maintains detailed information on each inventory item including part number, description, unit of measure, vendor and reorder data, item activity and complete information on current item costs, pricing and sales. Produces reports as follows: Physical Inventory Worksheet, Inventory Price List, Departmental Summary Report, Inventory Status Report, The Reorder Report and the Period-to-Date and Year-to-Date reports. Supplied in source code for Microsoft BASIC . . . . **$1,190/$30**

(T) **MAILING ADDRESS** — Keeps track of name and address information and allows the selective printing of this information in the form of mailing lists or address labels. Allows the user to tailor the system to his own particular requirements. User-defined format and print-out system uses a special format file which tells programs how to print the mailing list or address labels. Standard format files are included with system. Automatic sorting of data uses indexed file management routines which allow the name and address information to be sequentially retrieved and printed without file sorting. Supplied in source code for Microsoft BASIC . . . . **$790/$30**

• GRAHAM-DORIAN SOFTWARE SYSTEMS

(T) **GENERAL LEDGER** — An on-line system; no batching is required. Entries to other GRAHAM-DORIAN accounting packages are automatically posted. User establishes customized C.O.A. Provides transaction register, record of journal entries, trial balances and monthly closings. Keeps 14 month history and provides comparison of current year with previous year. Requires CBASIC-2. Supplied in source . . . **$995/$35**

(T) **ACCOUNTS PAYABLE** — Maintains vendor list and check register. Performs cash flow analysis. Flexible — writes checks to specific vendor for certain invoices or can make partial payments. Automatically posts to GRAHAM-DORIAN General Ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source . . . . **$995/$35**

(T) **ACCOUNTS RECEIVABLE** — Creates trial balance reports, prepares statements, ages accounts and records invoices. Provides complete information describing customer payment activity. Receipts can be posted to different ledger accounts. Entries automatically update GRAHAM-DORIAN General Ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source . . . . **$995/$35**

(T) **PAYROLL SYSTEM** — Maintains employee master file. Computes payroll withholding for FICA, Federal and State taxes. Prints payroll register, checks, quarterly reports and W-2 forms. Can generate ad hoc reports and employee form letters with mail labels. Requires CBASIC-2. Supplied in source . . . . **$590/$35**

(T) **INVENTORY SYSTEM** — Captures stock levels, costs, sources, sales, ages, turnover, markup, etc. Transaction information may be entered for reporting by salesman, type of sale, date of sale, etc. Reports available both for accounting and decision making. Requires CBASIC-2. Supplied in source . . . **$590/$35**

(T) **JOB COSTING** — Designed for general contractors. To be used interactively with other GRAHAM-DORIAN accounting packages for tracking and analysing expenses. User establishes customized cost categories and job phases. Permits comparison of actual versus estimated costs. Automatically updates GRAHAM-DORIAN General Ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source **$995/$35**

Sample Program Disk For Each Graham-Dorian Business Package. Specify Package **$45**

The sale of each proprietary software package conveys a license for use on one system only.

Prices F.O.B. Fort Worth, Tex. Shipping, handling and C.O.D. charges extra.

• MICROSOFT PRODUCTS

(M) **BASIC-80** — Disk Extended BASIC, ANSI compatible with long variable names, WHILE/WEND, chaining, variable length file records . . . . **$350/$25**

(M) **BASIC COMPILER** — Language compatible with BASIC-80 and 3-10 times faster execution. Produces standard Microsoft relocatable binary output. Includes MACRO-80. Also linkable to FORTRAN-80 or COBOL-80 code modules . . . . **$395/$25**

(M) **FORTRAN-80** — ANSI 66 (except for COMPLEX) plus many extensions. Includes relocatable object compiler, linking loader, library with manager. Also includes MACRO-80 (see below) . . . . **$500/$25**

(M) **COBOL-80** — Level 1 ANSI '74 standard COBOL plus most of Level 2. Full sequential, relative, and indexed file support with variable file names. STRING, UNSTRING, COMPUTE, VARYING/UNTIL, EXTEND, CALL, COPY, SEARCH, 3-dimensional arrays, compound and abbreviated conditions, nested IF. Powerful interactive screen-handling extensions. Includes compatible assembler, linking loader, and relocatable library manager as described under MACRO-80 . . . . **$750/$25**

(M) **MACRO-80** — 8080/Z80 Macro Assembler. Intel and Zilog mnemonics supported. Relocatable linkable output. Loader, Library Manager and Cross Reference List utilities included . . . . **$150/$25**

(M) **XMACRO-86** — 8086 cross assembler. All Macro and utility features of MACRO-80 package. Mnemonics slightly modified from Intel ASM86. Compatibility data sheet available . . . . **$300/$25**

(M) **PASCAL/M*** — Compiler generates P code from extended language, implementation of standard PASCAL. Supports overlay structure through additional procedure calls and the SEGMENT procedure type. Provides convenient string handling capability with the added variable type STRING. Untyped files allow memory image I/O. Requires 56K CP/M . . . . **$150/$20**

(M) **PASCAL/Z** — Z80 native code PASCAL compiler. Produces optimized, ROMable re-entrant code. All interfacing to CP/M is through the support library. The package includes compiler, Microsoft Compatible relocating assembler and linker, and source for all library modules. Variant records, strings and direct I/O are supported. Requires 56K CP/M and Z80 CPU. **$395/$25**

(M) **PASCAL/MT** — Subset of standard PASCAL. Generates ROMable 8080 machine code. Symbolic debugger included. Supports interrupt procedures, CP/M file I/O and assembly language interface. Real variables can be BCD, software floating point, or AMD 9511 hardware floating point. Version 3 includes Enumeration and Record data types. Manual explains BASIC to PASCAL conversion. Source for the run-time package requires Digital Research's MAC. Requires 32K . . . . **$250/$30**

(M) **CBASIC-2** Disk Extended BASIC — Non-interactive BASIC with pseudo-code compiler and run-time interpreter. Supports full file control, chaining. integer and extended precision variables, etc. . . . . **$110/$15**

(M) **BSTAM** — Utility to link one computer to another also equipped with BSTAM. Allows file transfers at full data speed (no conversion to hex), with CRC block control check for very reliable error detection and automatic retry. We use it! It's great! Full wildcard expansion to send ★.COM, etc. 9600 baud with wire. 300 baud with phone connection. **Both** ends need one. Standard and (M) versions can talk to one another. **$150/$5**

(M) (T) **SELECTOR III-C2** — Data Base Processor to create and maintain multi Key data bases. Prints formatted sorted reports with numerical summaries or mailing labels. Comes with sample applications, including Sales Activity, Inventory, Payables, Receivables, Check Register, and Client/Patient Appointments, etc. Requires CBASIC-2. Supplied in source . . **$349/$20**

GLECTOR — General Ledger option to SELECTOR III-C2. Interactive system provides for customized COA. Unique chart of transaction types insure proper double entry bookkeeping. Generates balance sheets, P&L statements and journals. Two year record allows for statement of changes in financial position report. Supplied in source. Requires SELECTOR III-C2, CBASIC-2 and 52K system . . . . **$250/$25**

(M) **TEXTWRITER III** — Text formatter to justify and paginate letters and other documents. Special features include insertion of text during execution from other disk files or console, permitting recipe documents to be created from linked fragments on other files. Has facilities for sorted index, table of contents and footnote insertions. Ideal for contracts, manuals, etc. Now compatible with Electric Pencil* prepared files. **$125/$20**

FORMATS AVAILABLE:

ORDERS MUST SPECIFY DISK SYSTEMS AND FORMATS:

(A) TRS-80 Model I (M) Keys Only
(B) TRS-80 Model II
(C) TRS-80 Model III (M) Keys Only
(D) HEATHKIT H89 (M) Keys Only
(E) NORTH STAR
(F) SUPER BRAIN QD
(G) STANDARD UNIMPLEMENTED

(M) Modified version available for use with CPM as implemented on Heath and TRS-80 Model I computers.

(T) For all (T) items listed above, the recommended system configuration consists of 48K CPM 2 full dize disk drives. 24 x 80 CRT and 132 column printer.

FMG's LIBRARY:

PASCAL USER MANUAL & REPORT
(2nd) Edition by K. Jensen and N. Wirth

- Tutorial Manual and Concise Reference Report for Both Programmers and Implementors
- Includes Helpful Examples to Demonstrate the Various Features of PASCAL

The book consists of two parts: the user manual and the revised report. The manual is directed to those who have some familiarity with computer programming and who wish to get acquainted with the PASCAL language. The report defines standard PASCAL, which constitutes a common base between various implementations of the language.

Stock No. #B21 — **Price** $9.95

Microcomputer Problem Solving Using Pascal
by Kenneth L. Bowles

- A Book Designed for Both College Courses AND Individual Self-Study
- Ideal for use with UCSD Pascal
- Includes Extensions to Standard PASCAL

This book is designed both for introductory courses in computer problem solving at the freshman and sophomore college level, and for individual self-study. It includes many examples and actually executable programs. It includes information on the necessary functions and procedures for handling graphics and strings.

Stock No. #B22 — **Price** $14.95

PROGRAMING IN PASCAL
by Peter Grogono

- An Excellent Introduction to One of the Fastest Growing Programming Languages Today
- Sections on Procedures and Files PLUS a Chapter on Dynamic Data Structures such as Trees and Linked Lists

The text is arranged as a tutorial, containing both examples and exercises to increase reader proficiency in PASCAL. Concepts are illustrated by examples, ranging from the Tower of Hanoi problem to circumscribing a circle about a triangle. PROGRAMMING IN PASCAL is sure to hold the reader's interest.

Stock No. #B23 — **Price** $14.95

PASCAL PRIMER Problem Solving

This book has three major goals:
- To introduce all aspects of the programming and problem solving process (includes problem specification and organization, algorithms, coding, debugging, testing, documentation and maintenance);
- To teach good programming style and how to produce a high quality finished product; and
- To teach the syntax of the PASCAL programming language.

Numerous examples are employed throughout the text. PASCAL is used as a vehicle to teach various aspects of programming techniques.

Stock No. #B24 — **Price** $18.95

BEGINNER'S MANUAL FOR UCSD PASCAL SYSTEM

- An Enlightening Introduction to UCSD PASCAL
- Demonstrates How to Use the UCSD PASCAL System and How to Program in PASCAL
- Includes Many Practical Examples of PASCAL Programs

This book is intended to be used as an introduction and reference manual for people just beginning to use the UCSD Pascal Software System. Whether you have never used a computer before or whether you are an experienced programmer who is unfamiliar with UCSD PASCAL, this book will provide a relatively easy, yet thorough, introduction to UCSD PASCAL.

Stock No. #B25 — **Price** $11.95

UCSD Reference Book

- A Reference Guide to the Complete UCSD PASCAL System
- Includes Information on Compiler Basic, Assembler and Editor
- Lists Actual P-Machine Codes

This reference book can be a valuable and time-saving guide to thorough information on the UCSD PASCAL system. The easy-to-read manual provides fast access to pertinent data.

Stock No. #B26 — **Price** $25.00

All that computer for $599.

The Imagination Machine, the personal computer from APF Electronics.

The Imagination Machine is more personal computer than you'd expect at $599.

The Imagination Machine is a superbly designed, expandable, user-programmable computer system... at $599.

No other personal computer on the market can touch it, at that price.

Read what it brings you:

First of all, The Imagination Machine has 9K RAM and 14K BASIC-IN-ROM. A full 53-key professional, typewriter keyboard. A high-resolution picture on your TV set, in eight colors. Fast loading (1500+ baud rate), built-in dual-track cassette deck, for APF's digitally recorded tape programs. Built-in sound synthesizer. And, even a built-in RF modulator, which is a $40 option on other computer systems.

All that, plus user-programmability.

We know sophisticated users aren't going to be satisfied forever using preprogrammed software. (Even though we offer a large library of educational, entertainment, home and business management programs.) So, we made The Imagination Machine user programmable, in both BASIC and MC6800 machine language. To simplify matters, we've just developed the first and only BASIC TUTOR course on cassette. With it, you can learn to program The Imagination Machine in BASIC, with hands-on training, right at the computer.

Some exceptional features.

The Imagination Machine has several unique features that can help you use your time at the computer more effectively.

For example, it stores programs **and** data on the same cassette tape. (With other computers, you have to read programs from one tape into the computer, remove the tape, put in another tape and store your data on the new tape.)

Another special feature is The Imagination Machine's unique keyword system, which simplifies BASIC programming. The machine has 24 different programs statements and commands printed at the top of the keyboard. You can enter these 24 into your program without retyping them every time you use them. Instead of typing out "PRINT," for example, you just press two keys and the word appears on the screen. The system helps prevent typing errors and can speed up entering programs.

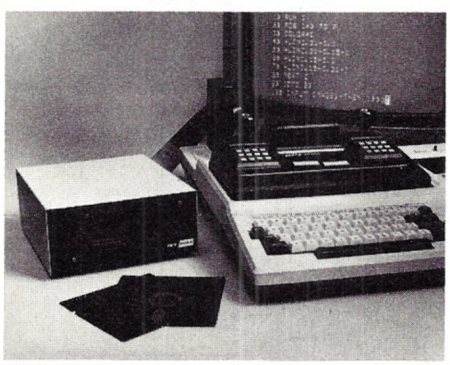
A third feature is Timed Response Monitoring, which automatically adjusts the computer's pace and level to your own. It makes "tutoring programs," for instance, easier and more interesting to follow.

And then there are The Imagination Machine's three graphic display modes: 1. Alpha numerics, mixed with low-resolution graphics in as many as eight colors. 2. High resolution — up to eight colors — 128 x 192 display. 3. High resolution graphics — up to four colors — with 256 x 192 display.

And expandability.

A personal computer that can't grow along with your growing requirements soon becomes obsolete. So, we designed The Imagination Machine to be expandable. By adding APF's optional "Expansion Box" and interface cartridges, you can hook up any compatible floppy disk or printer, or an additional 8K RAM memory cartridge.

Full mini-floppy system $995.



For small business and professional use, you may require a full mini-floppy system. In that case, order APF's System II. It includes The Imagination Machine, the "Expansion Box," floppy disk interface and 72K-byte, mini-floppy disk drive. All for just $995! No one can come close to that price.

You can't beat our prices or our guarantee.

If you can find a better personal computer system for the money, let us know. In the meantime, we stand by our statement: There is no other personal computer on the market that offers so much for so little. And if you order now, we'll even include our $19.95 APF Technical Reference Manual, with complete schematics, absolutely free.

Order The Imagination Machine directly from APF Electronics, with the assurance that if you are not completely satisfied, you can return it within 30 days of purchase for a complete refund. To order, or to learn the name of the dealer nearest you, call TOLL FREE 1-800-223-1264. New York residents call 212-869-1960. MasterCard and VISA accepted.

Price list:

System I, The Imagination Machine.	$599.
System II, Mini-floppy System (Includes The Imagination Machine, BB-2, and Mini-floppy Disk Drive).	$995.
BB-1. Expansion Box with RS232 cartridge.	$199.95
BB-2. Expansion Box with floppy disk interface cartridge.	$199.95
8K RAM memory cartridge.	$ 99.95
RS232 cartridge.	$149.95
Floppy-disk interface cartridge.	$149.95
Mini-floppy Disk Drive.	$399.95

$599. Manufacturer's suggested retail price.

apf electronics inc.
1501 Broadway New York, NY 10036

9

Two Jump-on-Reset Circuits For 8080 System Flexibility

Having trouble loading canned software because your system monitor is located in page zero? This hardware/software project could be just the solution you're looking for.

J. C. Hassall
H & H Enterprises
1201 Highland Circle
Blacksburg, VA 24060

If your system monitor resides in low memory (page zero), you have probably encountered the frustration of being unable to load some canned software bought at your local computer store or borrowed from a friend.

You are therefore faced with two alternatives: Get a listing and rewrite all addresses for wherever you have read/write memory, or move your system monitor.

The first alternative is out of the question for any but the most trivial programs. So how do you change the internal reset vector from page zero to somewhere else? You don't—but you can trick the CPU to vector to a location of your choice.

Background

Canned software assumes that your system has R/W memory available starting at page zero. A monitor located at page zero has some advantages, because the first instruction fetch cycle will go to address 000 000 (the first memory location in page zero), which is the first instruction in your system monitor.

The system monitor performs essential functions such as initializing the system and peripherals, and it contains commonly used subroutines (such as I/O, ASCII to binary and octal) and the system command decoder. You therefore want the system to jump there automatically upon power-up or system reset. Since a reset is automatically executed (using circuitry external to the CPU) on power-up, you can simply say that you want to be able to vector to the system monitor after a reset.

But what exactly is a reset command? It is an internal, non-maskable interrupt that clears the program counter, but leaves all other registers unaffected (at least with the 8080/8085). Since the PC is cleared upon reset, the CPU looks for the first instruction in memory at page zero. It is much easier to locate a programmable read-only memory (PROM) containing the system monitor starting at address page zero. Thus, immediately after reset, the CPU will commence fetching instructions without the need to vector the PC elsewhere in memory.

So the system functions beautifully until you try to load some commercial software that assumes R/W memory in page zero—right where your system monitor in PROM lives. Now what? You move your system monitor.

Several methods will accomplish this. You count clock cycles after reset and intercept the address bus; force the vector address onto the data bus and use an I/O line to disable the forcing function; intercept the address bus and disable the forcing function without software; or force a jump instruction onto the data bus. The last two methods are the simplest.

The components cost less than $5 for either circuit. That price is hard to beat.

As always, there are hardware/software trade-offs. The first circuit is hardware-intensive and uses no software. The second circuit is much simpler and uses less hard-

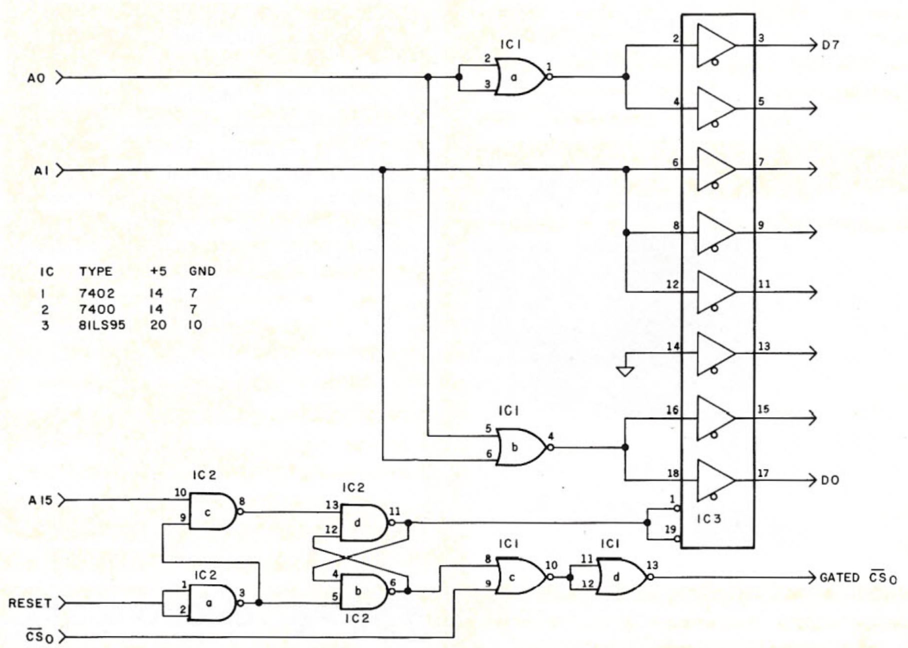


Fig. 1. Circuit 1, the straight hardware solution.

ware but has a software requirement. If your monitor cannot be modified, you must use the first circuit. However, if you can modify your monitor, the second circuit will probably be more to your liking.

I will use the convention of split octal in the following discussion of addresses. With that method, all 16 address lines can be represented by XXX YYY, where the eight X bits are the upper memory address byte (memory that would be addressed by the H register for the 8080/8085), and the eight Y bits are the lower eight bits of memory (memory that would be addressed by the L register for the 8080/8085). Thus, 000 000 is the first address in memory. It is also called page zero. Similarly, 004 000 is the first address in page one, and so on. Remember that page numbers are decimal, whereas the split octal numbers are octal. Thus, referring to Table 1, 370 000 would be the first address location in page 62.

Theory of Operation—Circuit 1

The program counter, after a reset, will fetch the first instruction by causing the CPU to output 000 000 on the 16 address lines and then do a memory read to fetch the contents of that memory lcoation. The second instruction is executed. Therein lies one of the keys to vectoring the CPU to your monitor address: Force the CPU to execute an unconditional jump instruction immediately after reset. With the 8080/8085, the sequence of instructions necessary for an unconditional jump are:

```
JMP
YYY
XXX
```

where JMP is the mnemonic for an unconditional jump, YYY is the low-order eight bits and XXX is the high-order eight bits of the address to be jumped to.

Now XXX and YYY need to be defined. For ease of programming, I like to have all R/W memory contiguous, so I don't have to jump over nonexistent memory or PROM. Since my monitor, less than 1K long, will be expanded, I have left 2K of memory space. Therefore, the monitor is located in page 62, with expansion to page 63 possible.

For ease of programming again, I have located the monitor at the page boundary (low-order bits are 000). Thus, the address to which I must vector is 370 000, where XXX = 370 and YYY = 000. Since we want to vector immediately after a reset command, and the CPU will execute a memory read (instruction fetch) cycle starting at 000 000 immediately after a reset, we need to have a memory address/instruction data byte correlation as follows:

address	data byte/mnemonic
000	JMP
001	000
010	370

You are rapidly approaching the final definition of the problem, but one other factor must be considered. Since you need R/W memory located at page zero, but you also need to have your vector instructions in the same memory space, they will fight on the data bus and produce garbage. You must therefore disable the chip select to page zero memory while you are vectoring the CPU, but re-enable the chip select immediately after executing the vector. You now have the full definition of the problem: how to jump unconditionally to 370 000 while disabling page zero memory and re-enabling it immediately after the vector.

Address A_1 A_0	Test Points X Y Z	Instruction	Instruction code (octal)	Data bus D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0
0 0	1 0 1	JMP	303	1 1 0 0 0 0 1 1
0 1	0 1 0	YYY	000	0 0 0 0 0 0 0 0
1 0	0 0 1	XXX	370	1 1 1 1 1 0 0 0

Table 3. Given the address inputs, the test points and data bus will follow this truth table.

Binary Address	Split Octal	Number Page Number
0 000 000 000 000 000	000 000	0
0 000 000 100 000 000	001 000	1
0 000 001 000 000 000	002 000	2
|	|	|
1 111 100 000 000 000	370 000	62
1 111 110 000 000 000	374 000	63

Table 1. Comparison of binary address with split octal and page number designation.

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

7402

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

7400

Input	Pin 1	Pin 19	Output
0	0	0	0
1	0	0	1
X	1	0	Z
X	0	1	Z
X	1	1	Z

81LS95

Input	Pin 1	Pin 15	Output
1	0	0	1
0	0	0	0
X	0	1	Z
X	1	0	Z
X	1	1	Z

8T97

Table 2. Truth tables for each chip, where X = don't care and Z = high impedance (Tri-state).

Circuit 1 Description

Referring to Fig. 1, you can see that only three chips are used. You will also see that only five input lines are required, while nine output lines are generated, eight of which go to the data bus. I have included truth tables in Table 2 for the chips used.

Table 3 will help you understand the circuit operation. Before you try to force the jump instruction onto the data bus, you must disable the chip select signal ($\overline{CS_0}$) for R/W memory in page zero. You must re-enable $\overline{CS}$ after the CPU vectors to the monitor. The sequence of events at reset is as follows.

When reset goes high, the address and data buses are Tri-stated, so A_{15} appears high to TTL logic. Since you will key on the low-to-high transition of A_{15} to disable the vector-forcing hardware, you need to eliminate the initial glitch at reset. The glitch is eliminated through IC2a and b.

After reset, A_{15} is gated through IC2c since reset is low. To re-enable $\overline{CS_0}$, the signal is gated through IC1c and inverted by IC1d. When reset is active (high), the output from IC2b is forced high, which causes the output from IC1c to go low, so gated $\overline{CS_0}$ goes high and the memory in page zero is deselected. The output from IC2d is forced low, which enables the Tri-state outputs of IC3. Thus, any inputs to IC3 will be transferred to the data bus.

With system memory disabled, the vector-forcing hardware will not have data bus contention, and so is free to force the jump instruction onto the bus. The next step is to gate the proper signals to the data bus at the proper time.

Gating the proper signals to IC3, depending upon the status of address lines A_0 and A_1 (A_0 being the least significant bit of the address bus), is the key to the circuit. The instruction code for the JMP instruction is 303 octal for the 8080/8085 (Table 3). If the op code for your processor is different from 303 octal, you will have to modify the connections and the truth table, but the principle applies to any system.

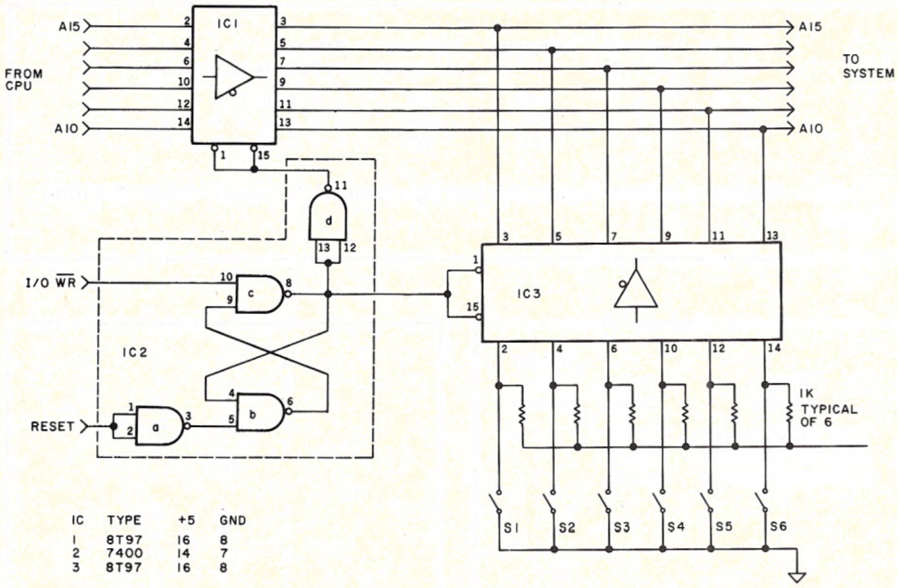


Fig. 2. Circuit 2, the hardware/software solution. Refer to Table 5 for alternate switch settings.

ory would stay selected as long as address lines A_{15} through A_{11} were forced high.

When A_0 and A_1 are both zero, data bits D_0 and D_1 are both 1, while D_0 and D_1 are both 0 for any combination of ones and zeros for A_0 and A_1. This is a NOR combination, so D_0 and D_1 are shunted together and wired to the output of IC1b. Since D_2 is always low, it is hard-wired to ground. Bits D_5, D_4 and D_3 all have the same state at the same time and have the same state as A_1, so they are shunted together and connected to the address line A_1. Bits D_7 and D_6 have the same state at the same time and have the inverse state of A_0, so they are shunted together, inverted through IC1a and connected to A_0. By sequentially stepping through Table 3, you can see the state changes of each data bit corresponding to each state change for address lines A_0 and A_1.

Immediately after executing the third instruction fetch cycle, the CPU will load 370 000 into the PC and jump to that address. The chip select $\overline{CS_0}$ for system memory in page zero is still disabled and must be re-enabled before you can use that 1K memory block. At the same time, you must disable the Tri-state output from the vector-forcing hardware, or you will have data bus contention. Since the CPU is going to output 370 000 on the address lines, you will use one of the upper order address lines (A_{15} was arbitrarily chosen) to retoggle IC2a and c. That forces IC2d output high, which deselects IC3 and forces IC2b low, which, when combined with $\overline{CS_0}$, will reselect page zero read/write memory.

Theory of Operation—Circuit 2

Circuit 2 (Fig. 2) operates more directly (in terms of hardware) than circuit 1, but has a software trade-off. Circuit 2 adds the monitor upper address byte to the address bus after reset. This assumes that the monitor resides at a page boundary. The address used for this circuit is the same as the previous circuit—370 000 octal.

As stated previously, the address on the address bus immediately after reset is 000 000. If you were now to force some address lines high, the CPU would think that it was accessing memory at location 000 000, but would actually be accessing another address. Specifically, you want to vector to address 370 000. The upper byte then must read 11 111 000 in binary. Thus, if the upper four address lines (A_{15} through A_{11}) were forced high, the CPU should think that it was accessing address 000 000, but memory in 370 000 would be selected. The memory would stay selected as long as address lines A_{15} through A_{11} were forced high.

Circuit 2 Description

This circuit has eight input lines and five output lines (Fig. 2). The low-to-high transition of the reset line upon reset enables IC3 through IC2a, b and c and disables IC1 through IC2d. IC1 acts as a buffer to isolate the CPU from the vector-forcing hardware. The outputs from IC3 could be wire-ORed with the address bus, but this is not a good digital design practice. The inputs to IC3 could be hard-wired or switched.

A DIP switch gives the ability to quickly change the address to which you want the CPU to jump after reset. Once IC3 is enabled and IC1 is disabled, they will remain as such until changed by an I/O $\overline{WR}$ pulse to IC2c. Here you encounter the hardware/software trade-offs mentioned earlier. You must cause the CPU to execute a jump instruction to the address where the monitor resides and then generate an I/O $\overline{WR}$ pulse to disable IC2 and enable IC1. Table 4 shows the software routine that must be added to your monitor to accomplish the task, and Fig. 3 is a tabulation of the memory addresses for the program, the status of the address bus from the CPU and the mnemonics of the program.

Note that the CPU thinks that it is accessing memory in page zero, but the address bus has been forced to page 62 by the vector-forcing function hardware. The JMP instruction causes the program counter in the CPU to be set to the address contained in the next two program data bytes. You don't need to execute an OUT instruction immediately after the JMP instruction.

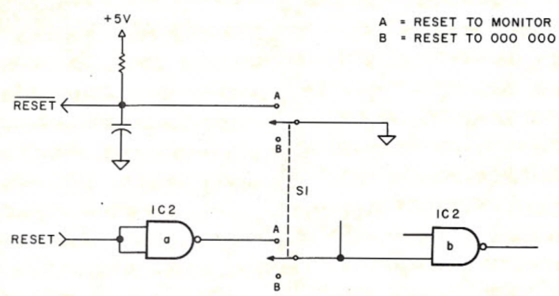


Fig. 3. This circuit will allow you to jump to either the monitor or address 000 000.

	Memory Location	Address Output from CPU	Mnemonic	Operation
Reset				
	370 000	000 000	JMP	; jump to
	370 001	000 001	003	; low byte (L reg.)
	370 002	000 002	370	; high byte (H reg.)
	370 003	370 003	OUT	; disable IC3
	370 004	370 004	XXX	; port address—doesn't matter

Table 4. What occurs on the address bus during execution.

For example, my monitor initializes the 8255 programmable peripheral interface (PPI) as follows.

```
MVI A  ; move immediate to A
202    ; this data byte
OUT    ; then output it to
007    ; I/O port 007
```

This sequence saves me from executing OUT XXX, then initializing the PPI: hence a memory savings of two bytes. Note that the port address byte XXX is not important; simply outputting to any port will suffice.

Switch Circuit

Note that DPDT momentary on the center off switch can be used to selectively reset to address 000 000 or to your monitor (Table 4). The analog reset components are illustrative only and will vary depending upon your system. You simply wire the switch in between inverter IC2a and gate IC2b.

When reset is low, the output from IC2a will be high. With the switch off or in position B, the input to IC2b will float high, having no effect on the circuit. The input to IC2b can only be pulled low by switching to position A, in which case reset will be high and IC2a output will be low.

Circuit Comparisons

Both circuits force the reset vector to some memory address other than 000 000. Circuit 1 requires no software modification. But circuit 2 requires only three chips. Circuit 1 must be physically located near the read/write memory in page zero. Circuit 2 should be physically located near the CPU.

I use circuit 2 because of the increased flexibility it gives me. But either way, if you've been having trouble with a program because your monitor resides in page zero, one of these circuits will solve your problem. ■

Acknowledgement

I would like to thank Linda Taylor for her assistance in the preparation of this article.

Switch Setting	Page Address	Octal Address
S_1 S_2 S_3 S_4 S_5 S_6	0	000 000
C C C C C O	1	004 000
C C C C O C	2	010 000
|	|	|
O O O O O C	62	370 000
O O O O O O	63	374 000

Table 5. By using this truth table, you can modify the jump-to-address for the monitor. Note that the switch closed corresponds to logic zero on the address bus.

Reader Service index—page 241

Mailing Label/Envelope Printer

This operator-oriented program for Micropolis Disk BASIC will assist even the most active and gregarious computer user with his paper correspondence.

Joel Shapiro
491 Kenilworth Court
Des Plaines, IL 60016

Many of my friends, business associates and clients are in organizations that require periodic mailings to their memberships. It was a good reason to adapt my existing mailing-list program (circa 1978) to my new data-base management system (see January 1980 *Microcomputing*, pg. 84).

This program, Labels, can be integrated with the data-base system by adding a call to the program from either Programs or Report. You can do this by adding the feature as a function to either menu and executing a PLOADG to Labels. My own preference is to call it from the Report program. None of the features of the data base system needs to be altered; the files are compatible for either way.

Program Features and Operation

Labels has the following features:

- It prints mailing labels 1, 2 or 3 across.
- The operator selects label width, height and spacing. It defaults to standard label dimensions and spacing.
- It prints name, title, company name, street address, city, state and zip code in five lines.
- It prints sorted information with the use of an index file and uses the sorting routine in the data base system.
- It ignores master file data that have been coded for deletion.
- It can print a partial master file by using upper and lower limits in the operator-selected control field as in the data base Report generator program.
- It does not leave open lines on label if data are not in the field. It packs label from the bottom up.
- A test routine is incorporated within the program for assistance in printer setup and adjustment.
- It permits the use of commercial self-stick labels or labels cut from your standard printout paper.
- It permits the use of continuous feed or manual feed envelopes. Spacing from the margin is operator-selected, as is the vertical spacing for continuous feed.

This program, like Database, is written in Micropolis Disk BASIC version 4.0. The system in use has a Z-80 CPU, 48K of RAM, Merlin video board, Micropolis dual disk drives (Mod II) and a DEC LA-36 printer with Accelewriter for 60 cps operation.

F$(X)	Elements 1–10 used for file setup and options.
G$(X)	File data read into this array. Also used within programs for parsing and other operations.
X$(X)	Stores field titles.
Z$(X)	Stores field length.
Y$(X)	Stores D, N or S field code and operator access code.
X(X)	Stores length of field (value Z$(X)).
Y(X)	Stores pointer for beginning of field.
B$(X)	Multipurpose data.
D%(X)	Tabs for printer.
A%(X)	Sorted record numbers from sort routine.

Arrays and their purpose.

F$(1)	File code.
F$(2)	Number of data fields.
F$(3)	Number of entries coded for deletion.
F$(4)	Fields and sequence for report.
F$(5)	Printer options.
F$(6)	File create date.
F$(7)	Special filename/purpose.
F$(8)	Data last update.
F$(9)	Fields for totals in report.
F$(10)	Reserved.
F$(11)–F$(30)	Available for programs.

F$ array details.

Labels is a self-contained program that can be called directly or from another Database system program. It combines the report generator and printer functions. When called, the program prompts the operator for specifics and prints labels and continuous feed envelopes automatically. When you use manually fed envelopes, the program will wait between printings so you can feed a new envelope.

The standard default label size is 3 1/2 inches wide and 15/16 inches high, for a width of 35 characters and a label of six lines. Thus the fields used in the master data file must be restricted to maximum lengths as follows:

Name	35 characters
Title	35 characters
Company Name	35 characters
Street Address	35 characters
City	25 characters
State	2 characters
Zip Code	5 characters

This will prevent truncation of the data strings and tabulation errors.

All fields are printed as individual lines with the exception of city, state and zip, which are concatenated into a single line, with the proper two-character space between the state and zip.

The above restrictions pertain to United States addresses only. For addresses outside the U.S., the city field should be

changed to 35 characters and used for the last line of the address. The state and zip fields should be left blank so the computer knows that the city field will be the last field printed.

The operator must ensure that the city field data does not exceed 25 characters when domestic addresses are intermixed with others. The domestic address will then be printing correctly. In both cases, the automatic stackup feature will function properly.

You can use any or all addressee fields—name, title and company name—but you must use at least one. Of these three fields, any missing field will be replaced by data from one of the others. If a name field (or data) is missing, the label will show title and company name. If the name and title are missing, the label will show just company name. Thus, the program can be used for either business or personal mailings, or both, without modification.

File compatibility with the Database program lets you use its features for all file creation and maintenance functions.

Uses

Using the Report generator in Database and the fields described herein, you can use a company name sort to see how much you are mailing to each company and to whom. This can save postage and eliminate duplicate mailings.

Most business mailings are sorted by zip code to take advantage of lower mailing rates. The sequence of printing depends on the type of sort used. If a multiple-level sort is used, you can sort for zip code and city and discriminate between small towns where a single zip code is used. An alphabetical sort can also be used if required.

Since the labels are printed horizontally across the page, they will be sorted in that format. Envelopes are printed in sorted order regardless of whether they are manually or automatically fed.

The use of upper and lower limits for a specified control field helps direct mailings to specific titles. Assuming a zip-code sort, a mailing can be directed to the presidents of companies by using the limits feature in the title field.

The control field does not have to be one of the fields selected for printing, as long as it appears in the file. Therefore, by proper coding in a nonprinting field and a multiple-level sort, mailings can be directed to a coded group of individuals or companies and sorted by zip code within that group. In a personal address file, a coding can be used to select addresses for Christmas cards. A business listing can be coded so that one group might buy a specific type of product, while another might include companies with the territory of a particular salesman.

I have incorporated a test routine that will let you print an outline of the printed label or address block to help with your printer setup. Because you will tend to use all available space, the printer setup must be as precise as possible.

I prepared a 14-entry master data file called Address to help demonstrate the program's features. The Report generator function of the Database program printed the complete master data file (Sample run 1).

Sample run 2 shows the file, sorted alphabetically, printed as three labels across. Note the proper formatting of the labels with regard to missing data in some of the fields.

Sample run 3 shows how to remove a field (in this case, title) from the label. A zip-code sort was used.

Sample run 4 used the control field limits to include only the range of items required in the printing. Compare this to Sample run 2 and note the difference.

As with the Report generator program in Database, the sensing of the word NAME as the first four letters of a field title will

```
NAME                                  TITLE                               CO. NAME
    STREET                                CITY                      ST  ZIP    DATE JOINED

                                      DIRECTOR OF SALES                   GAMES BY COMPUTER, INC.
    666 MERRYVILLE AVE                    MERRYVILLE                LA  70098  05/15/76

BAXTER, ROY T.                                                            BAXTER TOYS, INC.
    1 BAXTER AVE                          FERRYSTOWN                NJ  12112  05/05/75

HENRY, DR. TIMOTHY L.                 ADMINISTRATOR                       NORTH HOSPITAL
    45 WOOD TREE CIRCLE                   NORMAL                    FL  33447  07/18/78

JOHNSON, JOHN                         DIR. OF MATERIALS                   ACME DRY GOODS
    3232 W. MINSTER AVE                   FREMONT                   IN  74558  06/06/76

MATHIS, CHARLES M.                    PURCHASING AGENT                    BLOSSOM FLOWER CO.
    4554 PANSY WAY                        MARIGOLD                  IA  67789  12/16/76

SMITH, PAUL H.                        PRESIDENT                           SMITH COMPANY
    333 WEST 3RD STREET                   SMITHVILLE                CA  91104  08/14/78

ANDERSON, KENNETH                                                         ACE BRASS
    22 WEST 22ND STREET                   CAMBRIDGE                 MA  55739  11/12/77

BURTON, MR AND MRS
    34 7TH AVE                            CLEVELAND                 KS  22446  05/25/76

JACKSON, ANDREW                       VICE PRESIDENT                      JACKSON TOOL AND DIE
    388 JACKSON AVE                       JACKSON                   MS  99446  10/19/78

MARCUS, GARY L.                       SALES MANAGER                       TELEMAX, INC.
    887 ELM ST                            PETERSBURG                MS  66834  09/09/76

PRZYBLSKI, NORMAN T.                  GENERAL MANAGER                     FORD TILE AND FLOOR CO.
    665 CARPENTER AVE                     LASLOW                    ND  61111  05/26/77

THOMPSON, DARLENE                     BUYER                               WESTMONT INDUSTRIES, INC.
    WESTMONT AVE                          DENTON                    MD  99110  07/17/77

MURRAY, CHARLES K.                    PRESIDENT                           TERRANCE MFG
    43 LANE ROAD                          TORONTO ONT CANADA                       0/0/0

BORG, CATHERINE A
    567 WINDSOR LANE                      MONTREAL QUE CANADA                      0/0/0
```

Sample run 1.

DIRECTOR OF SALES GAMES BY COMPUTER, INC. 666 MERRYVILLE AVE MERRYVILLE LA 70098	KENNETH ANDERSON ACE BRASS 22 WEST 22ND STREET CAMBRIDGE MA 55739	ROY T. BAXTER BAXTER TOYS, INC. 1 BAXTER AVE FERRYSTOWN NJ 12112
CATHERINE A BORG 567 WINDSOR LANE MONTREAL QUE CANADA	MR AND MRS BURTON 34 7TH AVE CLEVELAND KS 22446	DR. TIMOTHY L. HENRY ADMINISTRATOR NORTH HOSPITAL 45 WOOD TREE CIRCLE NORMAL FL 33447
ANDREW JACKSON VICE PRESIDENT JACKSON TOOL AND DIE 388 JACKSON AVE JACKSON MS 99446	JOHN JOHNSON DIR. OF MATERIALS ACME DRY GOODS 3232 W. MINSTER AVE FREMONT IN 74558	GARY L. MARCUS SALES MANAGER TELEMAX, INC. 887 ELM ST PETERSBURG MS 66834
CHARLES M. MATHIS PURCHASING AGENT BLOSSOM FLOWER CO. 4554 PANSY WAY MARIGOLD IA 67789	CHARLES K. MURRAY PRESIDENT TERRANCE MFG 43 LANE ROAD TORONTO ONT CANADA	NORMAN T. PRZYBLSKI GENERAL MANAGER FORD TILE AND FLOOR CO. 665 CARPENTER AVE LASLOW ND 61111
PAUL H. SMITH PRESIDENT SMITH COMPANY 333 WEST 3RD STREET SMITHVILLE CA 91104	DARLENE THOMPSON BUYER WESTMONT INDUSTRIES, INC. WESTMONT AVE DENTON MD 99110	

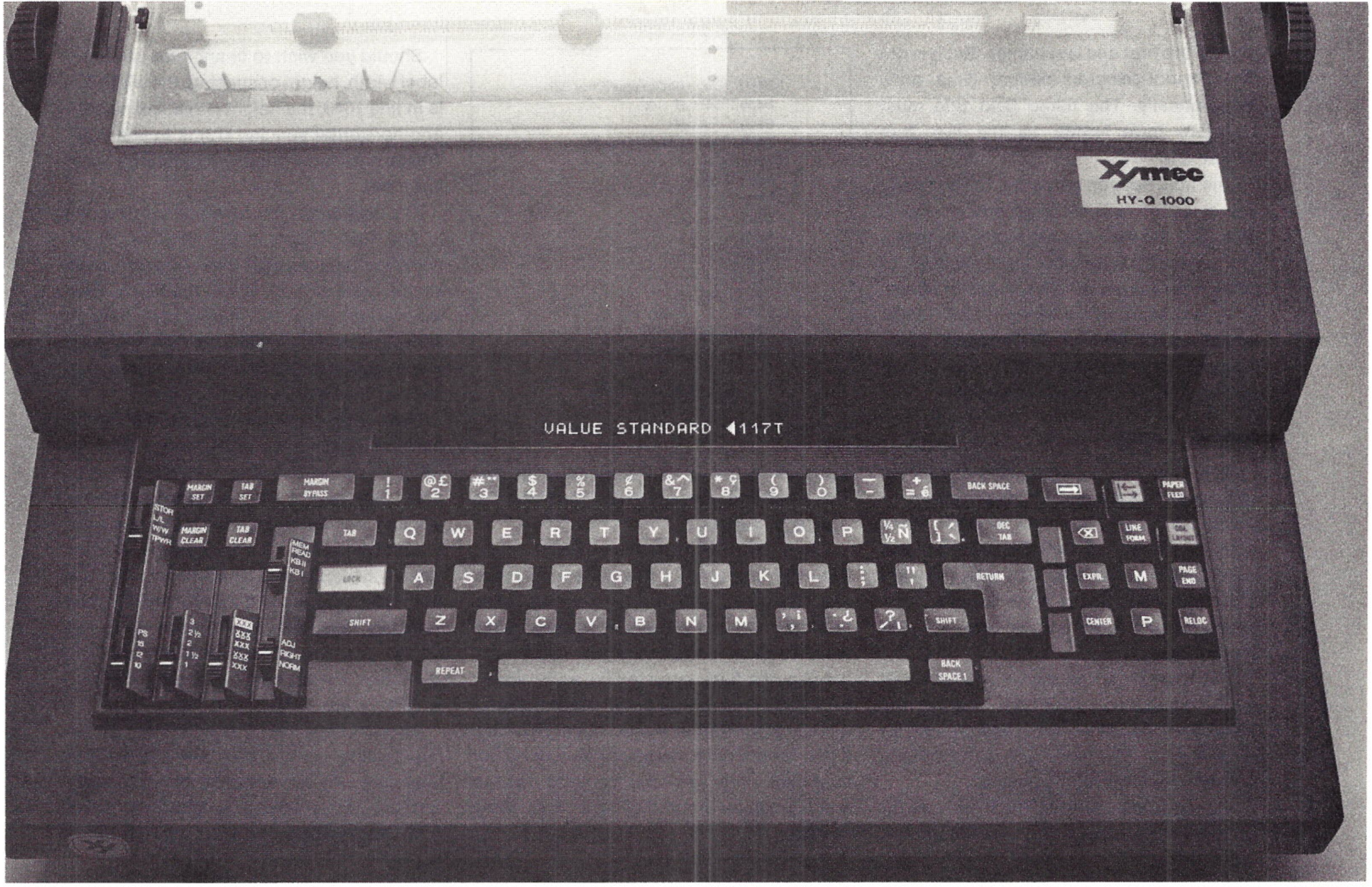
Sample run 2.

CATHERINE A BORG 567 WINDSOR LANE MONTREAL QUE CANADA	CHARLES K. MURRAY TERRANCE MFG 43 LANE ROAD TORONTO ONT CANADA	ROY T. BAXTER BAXTER TOYS, INC. 1 BAXTER AVE FERRYSTOWN NJ 12112
MR AND MRS BURTON 34 7TH AVE CLEVELAND KS 22446	DR. TIMOTHY L. HENRY NORTH HOSPITAL 45 WOOD TREE CIRCLE NORMAL FL 33447	KENNETH ANDERSON ACE BRASS 22 WEST 22ND STREET CAMBRIDGE MA 55739
NORMAN T. PRZYBLSKI FORD TILE AND FLOOR CO. 665 CARPENTER AVE LASLOW ND 61111	GARY L. MARCUS TELEMAX, INC. 887 ELM ST PETERSBURG MS 66834	CHARLES M. MATHIS BLOSSOM FLOWER CO. 4554 PANSY WAY MARIGOLD IA 67789
GAMES BY COMPUTER, INC. 666 MERRYVILLE AVE MERRYVILLE LA 70098	JOHN JOHNSON ACME DRY GOODS 3232 W. MINSTER AVE FREMONT IN 74558	PAUL H. SMITH SMITH COMPANY 333 WEST 3RD STREET SMITHVILLE CA 91104
DARLENE THOMPSON WESTMONT INDUSTRIES, INC. WESTMONT AVE DENTON MD 99110	ANDREW JACKSON JACKSON TOOL AND DIE 388 JACKSON AVE JACKSON MS 99446	

Sample run 3.

DIRECTOR OF SALES GAMES BY COMPUTER, INC. 666 MERRYVILLE AVE MERRYVILLE LA 70098	KENNETH ANDERSON ACE BRASS 22 WEST 22ND STREET CAMBRIDGE MA 55739	ROY T. BAXTER BAXTER TOYS, INC. 1 BAXTER AVE FERRYSTOWN NJ 12112
MR AND MRS BURTON 34 7TH AVE CLEVELAND KS 22446	DR. TIMOTHY L. HENRY ADMINISTRATOR NORTH HOSPITAL 45 WOOD TREE CIRCLE NORMAL FL 33447	ANDREW JACKSON VICE PRESIDENT JACKSON TOOL AND DIE 388 JACKSON AVE JACKSON MS 99446
JOHN JOHNSON DIR. OF MATERIALS ACME DRY GOODS 3232 W. MINSTER AVE FREMONT IN 74558	GARY L. MARCUS SALES MANAGER TELEMAX, INC. 887 ELM ST PETERSBURG MS 66834	NORMAN T. PRZYBLSKI GENERAL MANAGER FORD TILE AND FLOOR CO. 665 CARPENTER AVE LASLOW ND 61111
PAUL H. SMITH PRESIDENT SMITH COMPANY 333 WEST 3RD STREET SMITHVILLE CA 91104	DARLENE THOMPSON BUYER WESTMONT INDUSTRIES, INC. WESTMONT AVE DENTON MD 99110	

Sample run 4.



Now...
Intelligence beyond your CRT

HY-Q 1000

Xymec's HY-Q 1000™ Series brings microprocessor capability to printing. This fast, letter quality, intelligent printer/typewriter expands the capability of your computer or word processing system.

Xymec gives your system more of the features you want most:
- **L.E.D. display**
- **tractor feed**
- **RS232, IEEE-C779 interface**

Compatible with nearly all microcomputers, the HY-Q 1000™ Series prints **boldface** and reverse, centers titles, sets columns and right justifies. Your choice of three pitches and proportional spacing, with minimum throughput print speed of 20 cps. Changing print wheels or type styles takes but seconds.

Intelligent printing makes good sense to me: Send me more information and the name of my local dealer.

Name ____________________
Title ____________________
Company ____________________
Address ____________________
City ____________________
State ____________ Zip ____________
Telephone ____________________

Xymec, Inc.
17905 Sky Park Circle, Suite J
Irvine, CA 92714

Xymec™
The Value Standard in Printers

Xymec gives you more features than any other machine on the market, regardless of price!

Xymec intelligent printers won't create a service nightmare. Reliably constructed, the HY-Q 1000™ Series is built with serviceability in mind. And with service centers nationwide, maintenance is easy and convenient.

Give your computer system the intelligent edge. Join up with Xymec. For more information, send in this coupon today.

Xymec, Inc.
Subsidiary of Litronic Industries
17905 Sky Park Circle, Suite J
Irvine, CA 92714 (714) 557-8501

reverse the first and last names. But it is not an operator-selected option; it is done automatically. The name field title must therefore have NAME as the first four characters.

One final note on operating the program: The selection and stacking process takes time. But the features are worth the delay. You can anticipate about six seconds of delay between printing each set of labels. This is based on printing three labels across at a 4 MHz clock frequency. Less time is consumed when printing one or two across or when printing envelopes.

B$(1)	=	Backslash (Char 92)
B$(2)	=	D
B$(3)	=	N
B$(4)	=	Space (Char 32)
B$(5)	=	:
B$(6)	=	,
B$(7)	=	/
B$(8)	=	-
B$(9)	=	*
B$(10)	=	S
B$(11)	=	.
B$(12)	=	%
B$(13)	=	Date
B$(14)	=	Name
B$(15)	=	Amt.

B$ array details.

Program Details

As in the Database program, most lines up to line 1500 are subroutines common to almost all programs in this system. Once again, the functions defined in line 62 and statements in lines 302—331 are used with the Merlin video board. As in the other programs, you will have to substitute your own subroutines.

Line 302 brings the cursor to home and clears the screen. Line 330 reverses the video, and line 331 returns the system to normal video. The video reverse technique is used only for error messages.

The optional features such as control field limits and use of an index file are the same as in the other Database programs. The label and address block formatting is unique to Labels.

Lines 1500-1900 take care of information from the operator on the options required and the fields selected for printing. Note that the computer will ask for the field to access for each line to be printed. This allows the use of a fairly complex file as a master, for the program will use only those fields required for the labels or envelopes. This way, a file containing complete personnel or customer data can also be used for the mailing list without having a dual filing system.

Should you wish to change the standard label width, height or spacing, the width (T1) is in line 1640, the height (T2) is in line 1650 and horizontal spacing (T3) is in line 1660. The tabs for label printing are determined in line 1720.

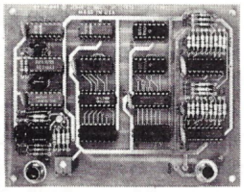
The standards for envelope printing are in line 1860. D%(1) is the variable for the number of spaces from the left-hand margin for the start of each line in the block. T2 is the variable for the number of vertical spaces between envelopes and correctly places the address block when using continuous feed envelopes.

Lines 2000-2240 take care of reading the file and loading the G$(X) array. G$(3) to G$(5) are used to hold up to three records of data. The record that is undergoing processing is in G$(1), and the parsed data are retained in G$(2). The subroutine in lines 1060-1061 takes care of extracting the data from G$(1) and placing them into G$(2). The selected field data are contained in the C%(X) array and are transferred into variable A for use with the subroutine. This occurs in line 2530.

After parsing, lines 2550-2670 assign the data to different variables. Lines 2630-2670 take care of concatenating the city, state and zip fields into a single line. Lines 2580-2620 provide the stacking feature so

MICRO MISCELLANY FROM JBE

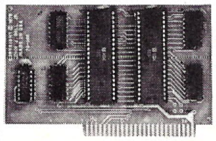
A to D D to A CONVERTER



The JBE A-D and D-A Converter can be used with any system having parallel ports, and interfaces with JBE Parallel I/O Card (see below). A-D conversion time is 20μS, D-A conversion time is 5μS. Uses include speech, music synthesizing, slow scan TV, and joystick or paddle control inputs. Uses single power supply (5V), see JBE 5V power supply below. Parallel inputs and outputs include 8 data bits, strobe lines and latches. Analog inputs and outputs are medium impedance zero to five volt range.

79-287 ASSM. **$79.95**
KIT **$59.95**
BARE BOARD **$29.95**

APPLE II PARALLEL INTERFACE



JBE Apple II Parallel I/O Card interfaces printers, synthesizers, keyboards, and JBE A-D and D-A converter and solid state switches. This interface has handshaking logic, two 6522 VIAs and a 74LS74 for timing. Inputs and outputs are TTL compatible.

79-295 ASSM. **$69.95**
KIT **$59.95**
BARE BOARD **$22.95**

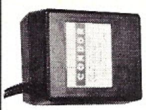
SOLID STATE SWITCH

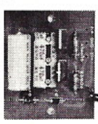
Control the world! Your computer can control power to your printer, lights, stereo and 120VAC appliances up to 720 watts (6 amps at 120VAC). Input 3 to 15VDC, 2 -13 MA TTL compatible, isolation - 1500V, non zero crossing, the switch comes in a 1 or 4 channel version and includes documentation for interfacing with JBE Dimmer Control (see below). The 1 channel version is also available professionally packaged.

79-282 1 CHANNEL ASSM. **$13.95** KIT **$10.95** BARE BOARD **$ 6.95**
1 CHANNEL PKGD. **$39.95**
79-282 4 CHANNEL ASSM. **$49.95** KIT **$39.95** BARE BOARD **$24.95**

POWER SUPPLIES

± 12 VOLT POWER SUPPLY

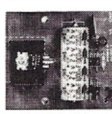




This 2x2½" power supply uses a wall transformer for safety and is protected against short circuit and thermal breakdown. It is rated at ± 12 V 120MA and can be used as a single 24V power supply at 120 MA. It is ideally suited to operational amplifier experiments.

80-161 ASSM. **$22.95**
KIT **$18.95**
BARE BOARD **$ 8.95**

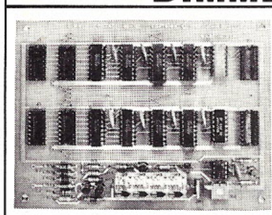
5 VOLT POWER SUPPLY



This 2¼x2½" 5V 500MA power supply is protected against short circuit and thermal breakdown and uses a wall transformer for safety. It operates JBE A-D and D-A converter, 8085 computer, 8088 computer & 6502 micro-microcomputer. Documentation is included.

80-160 ASSM. **$20.95**
KIT **$16.95**
BARE BOARD **$ 8.95**

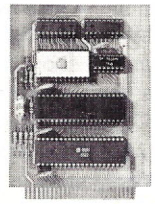
DIMMER CONTROL



The JBE Dimmer Control has 4 channels, 256 brightness levels, on-board power supply and four 8-bit parallel input ports (not latched). This board interfaces with the JBE Solid State Switch and Apple II Parallel Interface Card (documentation included).

80-146 ASSM. **$89.95**
KIT **$79.95**
BARE BOARD **$25.95**

6502 MICRO-MICROCOMPUTER



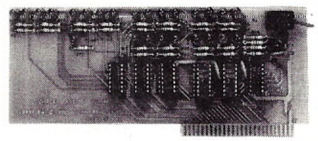
This JBE 3½x5" Micro-Microcomputer has the following:

- 1024 Bytes of RAM (two 2114s)
- 2048 Bytes of EPROM (2716)
- Uses one 6522 via (documentation inc.)
  - 2 8-bit bidirectional I/O ports
  - 2 16-bit programmable timer/counters
  - Serial Data Port
  - Latched output and input with handshaking logic.
  - TTL and CMOS compatible

The 6502 Microprocessor is particularly suited for control functions such as temperature control, burglar alarm, electric wheelchair, lights, etc. This Micro-Micro interfaces with the JBE Solid State Switch and A-D and D-A Converter and uses the JBE 5V power supply (see below). 2716 EPROM is available separately (not included in kit or assm. board). A 50 pin connector is included.

80-153 ASSM. **$110.95**
KIT **$ 89.95**
BARE BOARD **$ 24.95**

APPLE II DISPLAY BOARD



This handy little (3x7") board is ideal for teaching and troubleshooting. It has a run — stop, single step switch which makes identification of shorted lines between address or data-bits easy and shows single steps for teaching computer logic. The display board has 16 Address LEDS, 8 Data LEDS & 1 RDY LED. All lines are buffered.

80-144 ASSM. **$49.95**
KIT **$42.95**
BARE BOARD **$25.95**

BARE BOARDS

APPLE II EXTENDER BOARD

This is what you've been waiting for! The 3½x2½" Apple II Extender Board makes troubleshooting much faster and easier! Great for use with the JBE Apple II Display Board. 50 pin Apple connector is included.

80-143 **$12.95**

CRT CONTROLLER

This intelligent CRT Controller is completely contained on a 6x6½" printed circuit board. The design is based on an 8085A Microprocessor and an 8275 Integrated CRT Controller. It features the following:

- 25 Lines, 80 characters/line
- 5x7 Dot Matrix
- 8085 CPU
- Two 8185s
- Two 2716s (1 for software, 1 for user programmable character generator)
- Serial Interface RS232 and TTL
- Baud rates of 110, 150, 300, 600, 1200, 2400, 4800 & 9600.
- Keyboard Scanning System
- Uses +5V power supply and ± 12V power supply (both available from JBE — see above)

$39.95

8085 3-CHIP SYSTEM

State-of-the-art system using 3 IC's, an 8085, an 8156 and either an 8355 or 8755. The system has the following:

- 3 MHz 8085 CPU
- 256 bytes static RAM
- 2048 bytes ROM
- 38 parallel input/output lines
- 2 serial input/output lines
- Instruction set 100% upward compatible with 8080A
  14-bit counter/timer

$24.95

8088 5-CHIP SYSTEM

An 8086 Family microcomputer system using 5 IC's, an 8088 CPU, and 8284 clock generator, an 8155 RAM/IO/Timer, an 8755A EPROM/IO and an 8185 (1K x 8) Static RAM. This system has the following:

- 16-bit internal architecture
- Up to 1280 bytes of static RAM
- 2048 bytes of EPROM
- 38 parallel input/output lines
- 14-bit counter/timer
- Instruction set 100% compatible with the 8086

$29.95

SPARE PARTS

INTEGRATED CIRCUITS		CABLE	
6502	$ 9.95	Standard Dip Jumpers 16 Pin, 1 Ft. Length	$4.95
6522	$ 9.95	**CONNECTORS**	
2716 5 Volt	$29.95	50 Apple Connector	$5.95

JOHN BELL ENGINEERING ✓99

ALL PRODUCTS ARE AVAILABLE FROM: **JOHN BELL ENGINEERING**
P.O. BOX 338 • REDWOOD CITY, CA 94064 • (415) 367-1137
ADD 6% SALES TAX IN CALIFORNIA. FOR ORDERS OUTSIDE THE U.S.A., ADD 5% FOR SHIPPING AND HANDLING.

no blank lines will be printed if the field data aren't available.

The actual printing is done in lines 2710–2810. Line 2810 does the vertical spacing after printing for both envelopes and labels.

The test routine in lines 3000–3040 prints the outline of a label or address block to aid in printer setup.

The program will reload Database when completed. This is covered in line 2200. If you wish the program to load a program other than Database, this line will have to be changed. Remember that the disk for the program it is to load must be in drive 0.

Summary

The setup for the labels at a 35-character line length is crowded so I keep mine at a maximum of 33 characters in the file. I have not yet run short of line length, and this allows easier centering of the line on the label. The label I use is 3 1/2 inches by 15/16 inch and is obtainable from almost every office supply store in the Chicago area. The system is adaptable to any size you require.

The program has functioned well, and you shouldn't have any trouble with it. In fact, the only problem I can see is obtaining 1200 names for my mailing list to fill the system's capacity.

This program, with an updated Database system, is available on Micropolis Mod II disk for $20 postpaid. Purchasers of the original Data base system can have their Mod II disks updated for $10 and the return of their original disks. Send check and disk to: Bonjoel Enterprises, PO Box 2180, Des Plaines, IL 60018.■

Program listing. Labels program in Micropolis Disk BASIC, version 4.0.

```
61 SIZES(5,3,250,7150):MEMEND16RB7FF
62 DEFFAA=16R6B9:DEFFAB=16R6C2
63 DIM B$(15,4),F$(30,30),G$(5,250),X$(30,25),Y$(30,2),X(30),Y(30),Z$(30,3)
64 STRINGCHAR$(255):Y$="!"
65 RESTORE66:FORI=1TO15:READB$(I):NEXTI
66 DATA"\","D","N"," ",":",",","/","-","*","S",".","%","DATE","NAME","AMT."
68 DIMC%(7),C$(7,18),D%(3)
69 FORI=1TO7:READC$(I):NEXTI
80!
90 DIMA%(25),I$(3,40),J$(3,40),K$(3,40),L$(3,40),M$(3,40)
100 GOSUB302:K7=0:PRINT"LABEL AND ENVELOPE PRINTER":PRINT:PRINT"DO YOU WISH TO PRINT;"
110 PRINT" 1) LABELS"
120 PRINT" 2) MANUAL FEED ENVELOPES"
130 PRINT" 3) CONTINUOUS FEED ENVELOPES"
200 PRINT:PRINT"ENTER FUNCTION YOU DESIRE":INPUTA:IFA<1ORA>3THEN100
210 IFA=2THENK7=1
220 IFA=3THENK7=2
230 GOTO1500
302 POKE(16R6B8)=65:D=FAA:POKE(16R6B8)=41:D=FAA:RETURN
330 POKE(16R64E)=16R80:RETURN
331 POKE(16R64E)=16R00:RETURN
500!
510 GOSUB995:PRINT"ENTER NAME OF ";:IFA=1THENPRINT"MASTER FILE"
511 IFA=2THENPRINT"INDEX FILE"
512 PRINT"OR \ TO EXIT PROGRAM"
515 GOSUB999:INPUTN$:IFN$=B$(1)THEN2200
520 FORN9=0TO1:A$=MID$(STR$(N9),2,1):OPENAA$+":"+N$ ERROR540
530 N$=A$+":"+N$:T=FREETR(A):CLOSEA:RETURN
540 IFERR=4ORERR=7THENNEXTN9
550 GOSUB995:PRINTERR$:PRINT"STOPPED":PRINT"MAKE CORRECTION":GOSUB999:GOSUB997:GOTO500
560 GOTO510
570 GET2G$(1):I=1:K=1:A$="":IFG$(1)=""THEN2140
571 B$=MID$(G$(1),K,1):IFB$<>B$(1)THENA$=A$+B$:K=K+1:GOTO571
572 A%(I)=VAL(A$):A$="":K=K+1:IFI=25THENRETURN
573 IFK<LEN(G$(1))THENI=I+1:GOTO571
574 RETURN
603 GOSUB330:PRINT"ILLEGAL INPUT!  RE-ENTER":GOSUB331:RETURN
607 GOSUB998:PRINTTAB(10)"PROCESSING DATA":GOSUB999:RETURN
612 CLOSE1:GOSUB330:PRINT"DISK ERROR!":PRINTERR$:GOSUB331:GOSUB997:RETURN
830!   PARSE G$(R1)
831 IFI>10THENR1=3
832 IFI>20THENR1=4
834 J=LEN(G$(R1)):X=INDEX(G$(R1),Y$):IFX=0THEN838
836 G$=LEFT$(G$(R1),X-1):G$(R1)=RIGHT$(G$(R1),J-X)
838 RETURN
840!   PARSE G$ (TITLE, FIELD, CODE)
844 X=0:Y$(I)=RIGHT$(G$,2):G$=LEFT$(G$,LEN(G$)-2)
846 X=INDEX(G$,B$(1)):X$(I)=LEFT$(G$,X-1)
848 Z$(I)=MID$(G$,X+1,LEN(G$)-X)
849 RETURN
850!   LOAD X&Y ARRAYS WITH FIELD DATA
851 Y(1)=2:IFVAL(F$(2))<2THENRETURN
852 FORI=2TOVAL(F$(2)):Y(I)=Y(I-1)+VAL(Z$(I-1)):NEXTI
854 FORI=1TOVAL(F$(2)):X(I)=VAL(Z$(I)):NEXTI:RETURN
900!   COMPOSES DATE STRING (G$(2))
901 X=3:FORK=1TOLEN(G$(2)):A$=MID$(G$(2),K,1):IFASC(A$)<48ORASC(A$)>57THEN904
902 B$=B$+A$
903 NEXTK
904 G$(X)=B$:X=X+1:B$="":IFK<LEN(G$(2))THEN903
905 IFLEN(G$(5))>2THENG$(5)=RIGHT$(G$(5),2)
906 FORJ=3TO5:IFLEN(G$(J))<2THENG$(J)="0"+G$(J)
907 NEXTJ:G$(2)=G$(3)+B$(7)+G$(4)+B$(7)+G$(5):RETURN
924 FORI=1TO5:G$(I)="":NEXTI:RETURN
990 INPUT"( Y OR N )";A$:IFA$<>"Y"ANDA$<>"N"THEN990
991 RETURN
995 PRINTREPEAT$(CHAR$(13),25):RETURN
997 PRINT:INPUT"PRESS RETURN TO CONTINUE";A:RETURN
998 PRINTREPEAT$(CHAR$(13),9):RETURN
999 PRINTREPEAT$(CHAR$(13),7):RETURN
1000!   READS PARAMETER DATA AND SETS ARRAYS
1001 GOSUB302:GOSUB500:OPEN1N$ERROR1002:ATTRS(1)=3:GOTO1003
1002 GOSUB612:GOTO1001
1003 GOSUB1110:GOSUB302:GOSUB1030:GOSUB302:GOSUB850:RETURN
1030!   DISPLAY FILE HEADING
1032 PRINT"HEADING DATA FOR FILE - ";RIGHT$(N$,LEN(N$)-2):PRINT
1034 PRINT"CODE = ";F$(1):PRINT"NUMBER OF FIELDS = ";F$(2):PRINT"SPECIAL FILENAME = ";F$(7)
1036 PRINT"DATE CREATED = ";F$(6):PRINT"LAST UPDATE = ";F$(8):PRINT:IFG$(5)=""ORG$(5)="0"THEN1038
1037 PRINT"RECORD #5 MESSAGE":PRINTG$(5)
1038 IFVAL(F$(3))>1THENPRINTF$(3);" RECORDS CODED FOR DELETION"
1039 PRINT"DATA RECORDS IN FILE";SIZE(1)-5:PRINT"RECORDS REMAINING ON DISK ";T*16:PRINT:GOSUB997:RETURN
1050!   DISPLAY DATA (G$(1))
1052 FORI=1TOVAL(F$(2))
1054 G$(2)="":G$(2)=MID$(G$(1),Y(I),X(I))
1055 PRINTI;TAB(5)X$(I);TAB(25)G$(2)
1056 IFI=15ORI=2*15THENGOSUB997
1058 NEXTI:RETURN
1060!   PULLS OUT SPECIFIC FIELD FOR SEARCH
```

```
1061 G$(2)="":G$(2)=MID$(G$(1),Y(A),X(A)):RETURN
1100!   READ RECORDS 1-5 FROM FILE
1101 FORI=1TO5:GET1RECORDIG$(I):NEXTI:RETURN
1110!   READ G$(1-5)FROM FILE:LOAD ARRAYS
1112 GOSUB1100:GOSUB607:R1=1:FORI=1TO30:GOSUB834:F$(I)=G$:NEXTI
1114 R1=2:FORI=1TOVAL(F$(2)):GOSUB830:GOSUB840:NEXTI
1119 RETURN
1420!   REVERSES NAME (FIRST NAME FIRST)
1421 FORI5=LEN(G$(2))TOOSTEP-1:IFI5=0THENRETURN
1422 IFMID$(G$(2),I5,1)=CHAR$(32)THENNEXTI5
1425 G$(2)=LEFT$(G$(2),I5+1)
1427 G$="":E=INDEX(G$(2),B$(6)):IFE=0THENRETURN
1428 G$=LEFT$(G$(2),E-1)
1429 G$(2)=RIGHT$(G$(2),LEN(G$(2))-(E+1))+G$:RETURN
1500!   SET UP
1510 GOSUB302:A=1
1520 GOSUB1000:GOSUB924
1530 DATA"NAME","TITLE","COMPANY NAME","STREET ADDRESS","CITY","STATE","ZIP"
1540 GOSUB302:PRINT"ALL FIELD TITLES WILL BE DISPLAYED":PRINT"NOTE FIELD NUMBERS FOR:"
1550 FORK1=1TO7:PRINTTAB(10)C$(K1):NEXTK1:GOSUB997
1560 L3=1:GOSUB302:GOSUB1050:GOSUB997
1570 FORK1=1TO7
1580 GOSUB302:PRINT"ENTER FIELD # FOR ";C$(K1):PRINT"OR 0 IF NO FIELD":INPUTC
1590 IFC>VAL(F$(2))THENGOSUB603:GOSUB997:GOTO1580
1600 IFK1>3ANDC=0THENGOSUB603:GOSUB997:GOTO1580
1610 C%(K1)=C:NEXTK1:L2=L3-2
1620 IFK7>0THEN1860
1630 GOSUB302:PRINT"HOW MANY LABELS ACROSS SHEET":PRINT"ENTER 1 TO 3":INPUTL1:IFL1<1ORL1>3THENGOSUB603:GOSUB997:GOTO1630
1640 T1=35
1650 T2=6
1660 T3=1
1670 GOSUB302:PRINT"STANDARD LABEL SIZE IS:":PRINTT1;"CHARACTERS ACROSS AND"
1680 PRINTT2;"CHARACTERS DOWN":PRINT:PRINT"IS THIS O.K.":GOSUB990:IFA$="Y"THEN1720
1690 GOSUB302:PRINT"ENTER LABEL WIDTH (IN CHARACTERS)":INPUTT1
1700 PRINT:PRINT"ENTER LABEL HEIGHT (IN CHARACTERS)":INPUTT2
1710 PRINT:PRINT"ENTER SPACING BETWEEN LABELS (HORIZ)":INPUTT3
1720 D%(1)=1:D%(2)=D%(1)+T1+T3:D%(3)=D%(2)+T1+T3
1730 X4=0:GOSUB302:PRINT"YOU MAY ELECT TO PRINT YOUR LABELS":PRINT"WITHIN MINIMUM AND MAXIMUM LIMITS OF"
1740 PRINT"A SELECTED FIELD. DO YOU WISH TO USE":PRINT"THIS OPTION":GOSUB990:IFA$="N"THEN2000
1750 GOSUB1050:PRINT"SELECT FIELD FOR WHICH YOU WISH TO":PRINT"SET LIMITS":INPUTX4
1760 GOSUB302:PRINT"ENTER DATA FOR LOWER LIMIT OR \ IF YOU":PRINT"WISH TO DISREGARD LOWER LIMIT":INPUTL$:GOSUB1780:PRINT:PRINT
1770 PRINT"ENTER DATA FOR UPPER LIMIT OR \ IF YOU":PRINT"WISH TO DISREGARD UPPER LIMIT":INPUTU$:GOSUB1820:GOTO2000
1780 IFL$=B$(1)THENL$="":GOTO1800
1790 IFLEFT$(X$(X4),4)=B$(13)THENG$(2)=L$:FORI6=3TO5:G$(I6)="":NEXTI6:GOSUB900:L$=G$(2):RETURN
1800 IFLEN(L$)<X(X4)THENL$=L$+REPEAT$(CHAR$(32),X(X4)-LEN(L$))
1810 RETURN
1820 IFU$=B$(1)THENU$=REPEAT$(CHAR$(255),X(X4)):GOTO1840
1830 IFLEFT$(X$(X4),4)=B$(13)THENG$(2)=U$:FORI6=3TO5:G$(I6)="":NEXTI6:GOSUB900:U$=G$(2):RETURN
1840 IFLEN(U$)<X(X4)THENU$=U$+REPEAT$(CHAR$(32),X(X4)-LEN(U$))
1850 RETURN
1860 GOSUB302:D%(1)=40:T2=27:PRINT"ENVELOPE PRINTING":PRINT:PRINT"ENTER NUMBER OF SPACES FROM"
1870 PRINT"LEFT MARGIN FROM WHICH PRINTING":PRINT"WILL BEGIN (DEFAULTS TO";D%(1);")"
1880 PRINT:PRINT"JUST PRESS RETURN IF";D%(1);"IS O.K."
1890 INPUTD%(1):GOSUB302:IFK7=2THENPRINT"ENTER VERTICAL SPACING BETWEEN ":PRINT"ENVELOPES (DEFAULTS TO";T2;")":INPUTT2
1900 L1=1:GOTO1730
2000!   READ DATA ROUTINE
2010 M$=N$:A=2:CLOSE1:GOSUB302:PRINT"DO YOU WISH TO USE AN INDEX FILE":GOSUB990
2020 X8=0:IFA$="Y"THENX8=1:GOSUB2210:GOSUB500:OPEN2N$END2140:ATTRS(2)=3
2030 GOSUB302:OPEN1M$END2240:ATTRS(1)=3:OPEN3"*P"PAGESIZE66
2040 PRINT"SET UP PRINTER-DO YOU WISH TO TEST":GOSUB990:IFA$="Y"THENGOSUB3000:GOTO204(
2050 GOSUB607:GOSUB924:GETSEEK(1)=6:L=2
2060 IFX8=1THENGOSUB570:N=I:FORK1=1TON:IFA%(K1)<=5THEN2120
2070 L=L+1:IFX8=1THENGET1RECORDA%(K1)G$(L):GOTO2090
2080 GET1G$(L)
2090 IFLEFT$(G$(L),1)=B$(9)THENL=L-1:GOTO2120
2100 IFX4<>0THEN2220
2110 IFL-2=L1THENGOSUB2500
2120 IFX8=1THENNEXTK1:GOTO2060
2130 GOTO2070
2140 IFL<3THEN2170
2150 GOSUB2500
2160 IFK7=0THENENDPAGE3
2170 CLOSE3:CLOSE1
2180 IFX8=1THENCLOSE2
2190 GOSUB302:PRINT"DO YOU WISH TO PRINT ANOTHER FILE":GOSUB990:IFA$="Y"THEN100
2200 PLOADG"DATABASE"
2210 PRINT"PLACE DISK WITH FILE INTO DRIVE":GOSUB997:RETURN
2220 G$(1)=G$(L):A=X4:GOSUB1061:IFG$(2)<L$ORG$(2)>U$THENL=L-1:GOTO2120
2230 GOTO2110
2240 L=L-1:GOTO2140
2500!   PRINT ROUTINE
2510 FORJ1=1TO3:I$(J1)=" ":J$(J1)=" ":K$(J1)=" ":L$(J1)=" ":M$(J1)=" ":NEXTJ1
2520 FORJ1=3TOL:G$(1)=G$(J1):L2=0
2530 FORJ2=1TO7:A=C%(J2):IFA=0THEN2680
2540 GOSUB1061:IFX$(A)=B$(14)THENGOSUB1420
2550 IFLEFT$(G$(2),1)=B$(4)ANDJ2<7THEN2680
2560 IFLEFT$(G$(2),1)=B$(4)ANDJ2=7THENG$(2)=M$(J1-2):M$(J1-2)=" ":GOTO2580
2570 IFJ2>=5THEN2630
2580 IFI$(J1-2)=B$(4)THENI$(J1-2)=G$(2):GOTO2680
2590 IFJ$(J1-2)=B$(4)THENJ$(J1-2)=G$(2):GOTO2680
2600 IFK$(J1-2)=B$(4)THENK$(J1-2)=G$(2):GOTO2680
2610 IFL$(J1-2)=B$(4)THENL$(J1-2)=G$(2):GOTO2680
2620 IFM$(J1-2)=B$(4)THENM$(J1-2)=G$(2):GOTO2680
2630 IFJ2=5THENFORJ3=LEN(G$(2))TOOSTEP-1:IFJ3=0THEN2680
2640 IFJ2=5THENIFMID$(G$(2),J3,1)=B$(4)THENNEXTJ3
2650 IFJ2=5THENM$(J1-2)=LEFT$(G$(2),J3+1):GOTO2680
2660 IFJ2=6THENM$(J1-2)=M$(J1-2)+G$(2)+"  ":GOTO2680
2670 IFJ2=7THENM$(J1-2)=M$(J1-2)+G$(2):G$(2)=M$(J1-2):M$(J1-2)=B$(4):GOTO2580
2680 NEXTJ2:NEXTJ1:L=L-2
2690 IFK7=1THENPRINT"SET UP ENVELOPE FOR PRINTING AND":GOSUB997
2700 FORJ1=1TO5:FORJ2=1TOL
2710 ONJ1GOTO2720,2730,2740,2750,2770
2720 PUT3TAB(D%(J2))I$(J2);:GOTO2780
2730 PUT3TAB(D%(J2))J$(J2);:GOTO2780
2740 PUT3TAB(D%(J2))K$(J2);:GOTO2780
2750 IFLEFT$(L$(J2),1)<>B$(4)THENPUT3TAB(D%(J2))L$(J2);:GOTO2780
2760 GOTO2780
2770 IFLEFT$(M$(J2),1)<>B$(4)THENPUT3TAB(D%(J2))M$(J2);:GOTO2780
2780 NEXTJ2
2790 PUT3
2800 NEXTJ1
2810 FORJ1=1TOT2-5:PUT3:NEXTJ1:L=2:RETURN
3000!   TEST ROUTINE
3010 FORJ1=1TOT2:FORJ2=1TOL1
3020 IFJ1=1ORJ1=T2THENPUT3TAB(D%(J2))REPEAT$("X",T1);:GOTO3040
3030 PUT3TAB(D%(J2))"X"+REPEAT$(CHAR$(32),T1-2)+"X";
3040 NEXTJ2:PUT3:NEXTJ1:RETURN
```

SPECTACULAR
Offers

BASF "FLEXYDISK" Superior quality data storage medium, certified and guaranteed 100% error free.

5¼" or 8"* Diskettes 10/ $24

5¼" or 8" Vinyl Storage Pages 10/ $5

Write for quantity discounts
*Single sided / Single Density

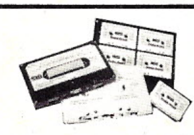
SFD CASSETTES "Super Ferro Dynamic" Using the finest Agfa PE 611 tape in a professional quality housing.

C-10 Cassette Sonic Weld Housing 10/ $7

Add 10¢ p/cassette for 5 screw housing

Cassette Album Page $1.89

Write for quantity discounts

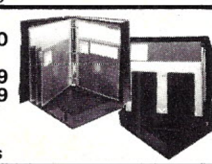


LIBRARY CASE 3-ring binder album, Protects your valuable programs on disks or cassettes. Fully enclosed and protected on all sides similar to Kas-sette storage box.

Library 3-ring binder $6.50

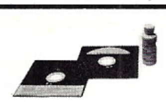
5¼"mini Kas-sette 10/$2.49
8" Kas-sette 10/$2.99

Write for quantity discounts



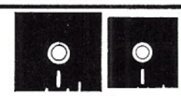
DISKETTE DRIVE head cleaning kits prevent head crashes and insure efficient error-free operation.

5¼" or 8" KIT
INTRODUCTORY PRICE
$19.50



HARDHOLE reinforcing ring of tough mylar protects your disks from damage.

8" applicator $4.00
5¼" applicator $3.00
8" mylar hardholes (50) . $8.00
5¼" mylar hardholes (50) $6.00



ABM PRODUCTS
631 "B" St.
San Diego, CA 92101
(714) 235-6602

273

VISA • MASTERCHARGE • MONEY ORDERS CERTIFIED CHECK • FOR PERSONAL CHECKS ALLOW 2 WEEKS • C.O.D. REQUIRES A 10% DEPOSIT • CAL. RES. ADD 6% SALES TAX MIN. $2 SHIPPING & HANDLING • MINIMUM ORDER $10 • SATISFACTION GUARANTEED OR FULL REFUND.

JOE COMPUTER Presents Exclusive Software:*

IT'S FOOTBALL SEASON!

Pro and College Football from SDL:
A TRS 80† translation of Ken Perry's popular Apple programs. These programs predict point spreads with unbelievable accuracy. They are heuristic and require about 10 minutes a week to record the weekend's results into the data base from your local newspaper. You may predict any game within seconds from the data saved on cassette or disk. Pro Football contains all 28 pro teams. College football contains an unbelievable data base of 78 teams! Each program comes with the entire 1979 season data file on cassette or disk. You can display each team's record of scores or won-loss record. A record of 4-5 weeks is required before predictions are effective so you're just in time! Pro or College Football will be shipped U.P.S. blue label the same day order is received. Order C.O.D. by phone. $1.00 blue label charge; C.O.D. fees added on.

Pro Football Cassette (32K TRS 80 Level II) **$21.95**
Disk **$26.95**
College Football Cassette (48K TRS 80 Level II) **$21.95**
Disk **$26.95**

Horse Race Handicapping!
Probability Handicapping Device 1 was written by a professional software consultant to TRW Space Systems. This is a complex program carefully human factored for easy use. It is a comprehensive horse racing system for spotting overlays in thoroughbred sprint races. Your computer will accurately predict the win probability and odds line for each horse based on your entries from the racing form. The next day overlaid horses can be spotted on the track tote board. The users manual contains a complete explanation of overlay betting plus much more useful information. The appendix contains a detailed tab run of a 100 consecutive race system workout showing an amazing 50% return ($.50 returned for each $1.00 flat wager). Includes many features such as error correction, bubble sort, line printer output, archiving, etc. The manual may be ordered seperately for perusal for $7.95 and credit. **PHD-1 users manual and cassette for: 8K Apple II Applesoft, 8K Challenger (specify 1P or 4P), TRS-80 16K Level II** **$29.95**
Apple or TRS-80 Disk **$34.95**

Brand new from SDL: Win At The Races another Ken Perry spectacular! This algorithm is based on a currently popular book representing the most ambitious multiple regression research on thoroughbred racing to date. The probable odds and win probability for each horse are displayed. Line printer output, error correction, descending sort; all the niceties!
Win At The Races cassette (32K TRS-80 Level II) **$34.95**
Disk (48K TRS-80 Level II) **$39.95**

THE BOOK for the Computerized Handicapper!
WINNING AT THE RACES by William Quirin Ph.D. Computer science has come to the rescue of the racing fan. This is the first major scientific study of handicapping available to the general public, detailing what the computer reveals about class, form, early speed, and more; plus special multiple regression computer systems. A Tom Ainsle — winners circle book.
Winning At The Races **$21.95**
+ $.75 P & H

Order now to get on our list and receive back issues free!
Phone Orders: Mike (213) 992-0514 Systems Design Lab (213) 374-4471

Make Checks payable to: **JOE COMPUTER** 247
22713 Ventura Blvd., Suite F, Woodland Hills, CA 91364

*Get on the Computers & Gambling Products mailing list for $3.00 & receive available back issues
Calif. res. add 6% tax. †TRS-80 is a Registered Trademark of Tandy Corporation

ATTENTION: TI-99/4 HOME COMPUTER OWNERS
HARVEY'S MUSICAL KEYBOARD:

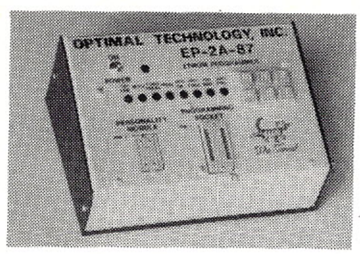
—Is a 633 line basic program which converts your keyboard into an organ with a 5 octave range. Program uses the sound generator built into the TI-99/4 computer.
—Is a program that assigns a note value to each key. The note plays when the key is depressed and continues to play until the key is released.
—Speed of play is acceptable, but slightly slower than a piano. Plays one note at a time to optimize speed of play.
—Play is in any one of 8 music keys. Program provides quick transition from one to any of the other music keys. One of the 8 music keys only has a 3 octave range due to program size.
—Accidentals cannot be played. This was necessary to allow a full 5 octave range. Not a serious limitation, very few songs require it.
—Entertaining and educational for most family members. Serious musicians who can play by ear should look into this.
—No special hardware requirements. The program cassette is all that you need.

Send $20 check or M.O. only, for the program on cassette and instructions, to:

177

J.H. HARVEY
p o box 4749
spartanburg, s. c. 29303

Model EP-2A-87
EPROM Programmer



The Model EP-2A-87 EPROM Programmer has an RS-232 compatible interface and includes a 2K or 4K buffer. During the ON-LINE mode, another computer can down-load to the buffer. Only two easy-to-implement commands are available to an external computer. (Load buffer and read buffer.)

In the OFF-LINE mode, the EP-2A-87 will program, verify, test buffer, and load the buffer from the EPROM socket. During the programming cycle, the EPROM is checked before programming to insure that it is erased and after programming it automatically verifies that programming is correct. Power requirements are 115 VAC 50/60 Hertz at 15 watts.

Part No.	Description	Price
EP-2A-87-1	Programmer with 2K buffer	$525.00
EP-2A-87-2	Programmer with 4K buffer	600.00
	Non standard voltage option (220 v, 240 v, 100 v)	15.00
PM-0	Personality Module, programs TMS 2708	26.00
PM-1	Personality module, programs 2708	26.00
PM-2	Personality module, programs 2732	31.00
PM-3	Personality module, programs TMS 2716	26.00
PM-4	Personality module, programs TMS 2532	31.00
PM-5	Personality module, programs 2716, TMS 2516	16.00
PM-6	Personality module, programs 2704	26.00
PM-7	Personality module, programs 2758, TMS 2508	16.00
PM-8	Personality module, programs Motorola MCM68764	34.00
MS-XX	Disk driver software	27.50

Optimal Technology, Inc.
Blue Wood 127
Earlysville, Virginia 22936 29
Phone (804) 973-5482

Dial-up Directory

Computers and communications learn to live together.

Frank J. Derfler Jr.
PO Box 691
Herndon, VA 22070

If you've been following these columns, you have some idea of the ways in which data communications can be used. But you haven't seen anything yet.

Here are some recent news items:

- The Source is joining with Cox Cable Communications to provide Source services to as many as 10 million homes.
- GTE is starting its own electronic mail service on Telenet (5 cents per minute on off-time rates).
- The Source will be providing service to public libraries around the country.
- Southern Pacific Communications wants to put up their own satellites by 1983 for low-cost data transmission services across the nation.
- Amdahl Computers is moving into data communications.
- Tymet and Satellite Business Systems are going to provide low-cost, high-speed data transmission circuits.

These items should give you some idea of the explosion taking place in the data communications field. You and your micro are sitting on the edge of a stream tossing in pebbles. Meanwhile, the dam has burst above you.

Alabama		
Birmingham	205-945-1489	ABBS.
Arizona		
Phoenix	602-866-0258	ABBS.
Phoenix	602-957-9282	7 PM–10 AM daily, 24 hrs., Sunday.
California		
Inglewood	213-673-2206	Not 24 hours.
Santa Monica	213-396-3905	ABBS.
Florida		
Miami	305-261-3639	Byte Shop ABBS.
Georgia		
Augusta	404-793-1045	ABBS, software exchange.
Illinois		
Macon County	217-429-5505	6 PM–6 AM, 24 hrs., Sunday.
Arlington Hts.	312-255-6489	9 AM–9 PM, 24 hrs., Sat. & Sun.
Michigan		
Detroit	313-357-1422	Michigan Apple-Fone.
Washington		
Tacoma	206-937-0444	Apple Bin ABBS.
Vancouver	206-244-5438	Apple Crate II ABBS.
Elma	206-482-5590	6 PM–12 AM, 24 hrs., Sat. & Sun.
Seattle	206-246-8983	Message system.

The following list of dial-up systems is taken from my file of over 200 phone numbers (most of them bad). I have been on each of these systems at least once, but that is my only guarantee.

30 Years Ago

Once upon a time, we only had telephones. Then computers entered the scene.

In 1950, the United States set up IBM's Semi-Automatic Ground Environment (SAGE) system for the air defense of North America. Radar data were sent to the SAGE computers from remote sites in digitized form over dedicated telephone lines (at 1200 baud). This is the earliest joining of communications circuits and data processing on a major scale.

Both computers and communications continued to grow, but with little acknowledgement of each other's existence. Computers used telephone lines to support remote job entry terminals (a way to transmit punch card data), and the telephone company used computers to send out bills. This situation continued through the 60s.

The integrated circuit/digital revolution of the 70s linked the two. The world of telecommunications grew like topsy.

Computers needed to exchange data, and terminals needed access to computers. The phone system was the best way to provide that access and exchange. The phone

companies (we will call them carriers now) needed to do phone call switching and provide inexpensive dialing and billing. Computers switched calls for computers so computers could talk to computers, terminals and (more recently) people. (The first time you get the synthesized voice that reads the out-of-service number is spooky.)

But many people were afraid to recognize that the systems were becoming more uniform. We had communications computers and computers that communicated, but they were regulated and managed in different worlds.

Not the least of the slow thinkers was the federal government. We all probably know about the power of the Federal Communications Commission (FCC) to regulate the common carriers. You may not know that the data processing companies have been struggling under some regulations, too. They have been bound by the FCC, the Commerce Department and others.

Some of these regulations protected certain market areas. Others opened various selected doors to competition. Often, regulators seemed to be rushing off in many different directions at the same time. They had no cohesive regulatory policy.

Meanwhile, the line between communications and computers practically ceased to exist. Several companies, such as Xerox, were getting more and more into what had been considered communications. Others, such as GTE, were taking some bold steps in the area of information processing.

The Congress and the bureaucracy agonized over regulatory reforms throughout the last of the 70s with no great success. Finally, in a surprising move, the FCC came to the rescue. In their April 7, 1980, Second Computer Inquiry, they effectively deregulated the data communications marketplace.

This is a brave and wise move that should be welcomed by anyone interested in computers (big or small) and their use at every level of society. Unfortunately, the White House and Congress seem to think the FCC stepped out of bounds by deregulating the industry. We can only hope the wisdom of the FCC move prevails.

I'm not suggesting that your system is going to be compatible with all these new developments, but the things you are learning will be. Every new piece of software, every new technical twist in communications, every new application you come up with better prepares you for the upcoming data communications explosion.

The industry is crying for people who understand data communications. Exchanging a few programs on an ABBS doesn't make you an expert, but it certainly should give you a new perspective on what you might study to become an expert.

Robert Angliss, the executive vice-president of RCA Global Communications, recently talked about a new career area he called "movers of information"—a combination of communications and traditional computer operations. Learn both the technical and applications side of information moving, and you'll be set for the long haul.

B	=	Bulletins. Reprints bulletins.
E	=	Enter a message into system.
G	=	Goodbye. Leave system (hangup).
H	=	Help with various functions.
I	=	Information about system.
K	=	Kill a message from the files.
M	=	Message alert. Messages for you?
O	=	Other systems current summary.
Q	=	Quickscan of message headers.
R	=	Retrieve a message from the files.
S	=	Scan of message headers.
SR	=	Selective message retrieval.
T	=	Time, date and connect time.
U	=	User modifiable system functions.
X	=	Expert user mode (on/off toggle).
Z	=	Continue message entry after abort.
?	=	Prints list of commands.
*	=	Flagged message memory retrieval.
ALT	=	Switch msg files (toggle).
DOS1	=	Article on Apple DOS, part 1.
DOS2	=	Article on Apple DOS, part 2.
DOS3	=	Article on Apple DOS, part 3.
MIND	=	Article by Dr. David Hoy.
TEST	=	Modem continuous test loop.
ASCII	=	Printer-formatted ASCII character chart.
USERS	=	File of system users/interests.
UPDATE	=	Messages from Arpanet.
UPDATE2	=	More messages from Arpanet.
CAL1980	=	1980 calendar/printer format.
NEWCALL	=	Information for new callers.
AUTOLOG	=	Change your autolog defaults.
RESPONSE	=	User responses to UPDATE.
GENERAL14	=	Download programs.
GENERAL15	=	Upload programs/files.

This list of system commands is a little longer than most, but it gives you a good idea of what can be done. It is from Bill Blue's Peoples' Message System in Santee, CA (714-449-5689).

Table 1. Command summary.

The Opening Moves

I have resisted explaining how to use various electronic bulletin systems because they are simple and because anything I print today may be different tomorrow. But many people are afraid of either making fools of themselves or of somehow damaging the system.

Computer Bulletin Board, Forum-80 and Apple Bulletin Board make up the vast majority of systems. My list shows more ABBS systems than anything else. I said last month that Bill Abney (Forum-80 founder) will provide instructions for Forum-80 systems if you send him a large envelope with double first-class postage. So let's look at an ABBS and take away the mystery.

The basic ABBS consists of an Apple II computer with 48K of memory, AppleSoft BASIC in ROM, two disk drives and a D.C. Hayes modem board. Some systems have augmented the disk memory—all the way up to ten megabytes—but the operation is the same. The modem board is connected to a phone line and waits for the phone to ring.

At your end, your computer or terminal should be set for full duplex, eight-bit words, no parity and one stop bit at 300 baud. (Seven-bit words and even parity will work, too.) Your modem should be in the originate mode with full duplex selected. If you are using a computer as a smart terminal, you should instruct the software to get ready to communicate in full duplex.

A work session with an electronic bulletin board or message system can be divided into four periods: sign on, bulletins and introduction, message exchange and sign off.

First, you dial the phone number of the ABBS. If you are using an acoustic modem, such as the Novation CAT, you listen for the phone to ring and for the other end to answer. An ABBS will normally answer after the first ring. If you get three rings and no answer, something is wrong and you should hang up. (Did you misdial?)

When the ABBS answers, you should hear a steady answer tone. You must immediately put the phone in your modem's cradle. The CAT takes about 1.5 seconds to

recognize the answer tone and send its own originate tone. This gives you about 8.5 seconds to get the phone to the CAT before the ABBS considers your call a wrong number and hangs up.

After the ABBS recognizes your tone, it will transmit "TYPE CARRIAGE RETURN (CR)" twice. Respond each time with a carriage return. This allows the ABBS to determine the baud rate.

Some ABBS versions may also ask you if you have a user ID. These systems recognize regular users when they sign on and tell them if they have any messages addressed to them. Reply "no" to the "USER ID?" question if you see it.

If you become a regular user, you and the system operator can get together and establish your ID. The system will request your name, location and phone number for logging. The ABBS will ask you to double-check the information and log you to the disk. This may take a few seconds, so have patience; the information is important to the system operator.

That completes your sign on. The second phase of system use, bulletins and introduction, begins.

The ABBS will probably print a welcome message and provide you with some bulletins. These may include system changes, news of club meetings or operating hours. This will be followed by a list of commands the system will respond to. The list may be long, but you are not concerned with most of the options. I have provided a typical command summary (Table 1) but will concentrate on only S, R, E and G. Finally, you will be presented with the following list:

(A, B, D, E, G, H, K, L, N, P, Q, R, S, T, V, W, X, ?)?

This is the command line. You start all new functions from this point and actually start using the system.

The ABBS will wait for you to input data. If this is your first time on, I strongly suggest that you send the "?" command for a complete explanation of the commands. Scan the messages available (command-S) for the message headers. It will show who the message is from and to, the date and the subject.

You may be in for a little surprise. Some systems have several hundred messages on file. Large systems will wisely ask you if you want to limit your selection. Look through about 100. Write down the numbers of those that interest you. (Many systems have automatic flagging of messages you later want to retrieve.)

After the message scan, you will return to the command line. Enter a command R for retrieve. You can then retrieve the full messages you want to read by message number. Various subroutines provide prompts and help within each of the command functions.

When you return again to the command line, you may wish to enter a message of your own with the E command. Again, you will be guided through each step of message entry. Finally, when you are done, enter a command G for "goodbye." Always sign off with G. If you don't, the system will be unavailable to other users for about eight minutes, or until it is convinced you are gone. If you are confused or make an error, you can get back to the command line by hitting a control-K. The system will quit whatever command it is on and return you to the command line.

Some people are afraid they might screw up the bulletin board system. But this is tried by experts every day, and few succeed. Many smart (but twisted) people make system-busting their hobby. System-busters were sometimes successful when bulletin board software was new, but the standard systems are now practically immune to sabotage. So you are not going to crash the system with a few mistaken commands.

Bulletin board systems of all types are out there to serve you for free. They are helpful and practically indestructible. Give them a try.

Forum	Everett, WA	(3.0)	206-334-7394	
Forum	Leavenworth, KS	(3.0)	913-651-3744	(Educational)
Forum	Monmouth County, NJ	(3.0)	201-528-6623	
Forum	Orlando, FL	(3.0)	305-862-6917	Evenings—W/E
Forum	Orange County, CA	(3.0)	714-952-2110	
Forum	Seattle, WA	(3.0)	206-723-3282	
Forum	Tulsa, OK	(3.0)	918-224-5347	Evenings—W/E
Forum	Westford, MA	(3.0)	617-692-3973	

More Forum-80 systems to add to last month's list. The number in parentheses refers to the version of software in use.

Try Me

If you sell items for data communications, run a system or have had some interesting experiences, your comments and news items are welcome. Send paper mail to the address at the beginning of the article (include a stamped envelope if you want a reply) or address electronic mail to TCB967 on The Source, 70003, 455 on MicroNet, or the AMRAD CBBS (703-734-1387). ■

✓ Reader Service index—page 241

The 1802 EPROM Board

Give your Elf II easy access to monitors and programs.

Dan Rubis
PO Box 402
St. Clair Shores, MI 48080

Since the inception of the RCA 1802 microprocessor chip, the software support has been fragmentary. More than a dozen different computers based on the 1802 have their own operating system and utility programs.

My system consists of an Elf II with Giant Board and 4K of static memory. To take full advantage of the machine, I realized that easy access to monitors and programs from various sources would be a plus.

I also wanted to be VIP-compatible, since numerous programs are already developed by RCA for their VIP computer. To do this, I needed to wire up a scanning keyboard similar to the one used by the VIP.

Also, the Chip 8 interpreter had to be loaded at pages 00 and 01, and the VIP operating system needed to be located at page 80 in memory. A page is 256 bytes, and there are 256 pages of memory addressable by the 1802 microprocessor with 16 bits for addressing. As an alternative, the operating system can be relocated from page 80 to some convenient place in your RAM memory, but not without changing the program.

Relocating the operating system is marginal at best. Besides using up valuable RAM space that could be better applied to user programs, user program bugs have a bad habit of wiping out the RAM-loaded operating system. The operating system must then be loaded several times while developing machine-language or VIP Chip 8 programs. The constant wear and tear on my cassette tape recorder was in itself enough to justify an EPROM board.

The same goes for Netronics' Elf Bug, which is a real asset for helping to debug your machine-language programs. It is relocatable to any place in memory without any change. But those prolific programming bugs can wipe out Elf Bug, too.

I also saw the need for EPROMing some of my personal utility programs such as clear memory, block move, RCA's standard call and return subroutines, RS-232 hex dump and video refresh routines. You might want to dedicate the entire EPROM to a single program or high-level language such as floating point subroutines, or wire up 4K and have Tiny BASIC on line.

Why use the higher-priced 2716 over the 1702 or 2708 EPROMs? Although the 2716 is two times the price per bit of ROM memory compared to the 2708, it runs on a single power supply of 5 volts. This saves the cost of a −5 volt and a +12 volt power supply required for the 2708 EPROM.

One of the benefits of the 1802 microprocessor series is its low power consumption CMOS design. My system draws only a little over 1 amp of current, most of which is eaten up by the static memory. The 2716 consumes 50 percent less power in the active mode and 75 percent less power in the standby mode than the 2708, which does not have the standby mode feature. This is another way to reduce the power requirements of your system.

As of this writing, 2708s cost $6, and 2716s cost $19. Nine months earlier retailers were asking as much as $60 for a 2716. With this trend in pricing, it could drop to the $8 to $13 range in another six months.

Thus, you can have 2K of 2716 EPROM memory locatable at any 2K boundary in memory for the cost of two 2708 EPROMs, and have the capability of easily upgrading the board to 4K in the future.

Design

The National Semiconductor six-bit bus comparator is the heart of this two-chip design (Fig. 1a). There are six exclusive-NOR gates; Fig. 1b shows the truth table for this gate.

Note that the exclusive-NOR gate goes to logic 1 (high) if, and only if, both the T and B inputs are logically identical. Outputs from six of these exclusive-NOR gates are inputs to an AND gate. The AND gate needs all inputs logically high before its output goes high. Thus, you have six comparators in one neat package. Each has a pair of inputs that have to be logically identical before the output of the device goes active low. Now you have the tool available to address the EPROM.

Addressing

The address bits are labeled A0 through A15. The decimal number located above each bit is the power of each binary digit in decimal (Table 1). Looking at the pin-out for the 2716 (Fig. 1c), you will see that it requires address bits A10 through A0 for its addressing.

Table 1 shows 1024 directly over address bit A10. Therefore, this EPROM uses 2048 unique memory locations. Watching bits A11 through A15 will allow you to select the EPROM when needed, and bits A10 through A0 will allow accessing the individual bytes from the EPROM.

The 8131 bus comparator steps in to watch the desired bits. By forcing the T inputs of the comparators either high or low (depending on the bit pattern for the desired pages) and connecting the B inputs to the address bus, you can watch the bus for the corresponding bit pattern.

The T inputs are forced high or low by applying a high to the inputs with a resistor connected to the +5 volts of the circuit (see the schematic in Fig. 2) and connecting a DIP switch between the T inputs and ground. Depending on whether the switch is open or closed, the T inputs will be either high or low.

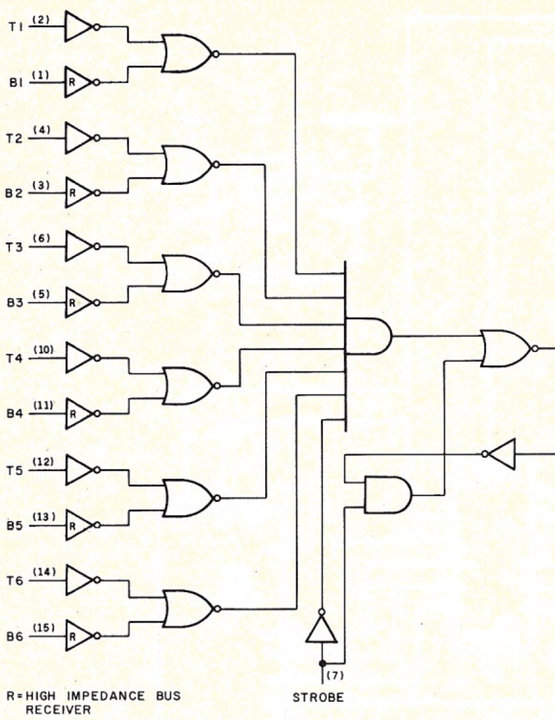
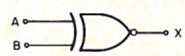
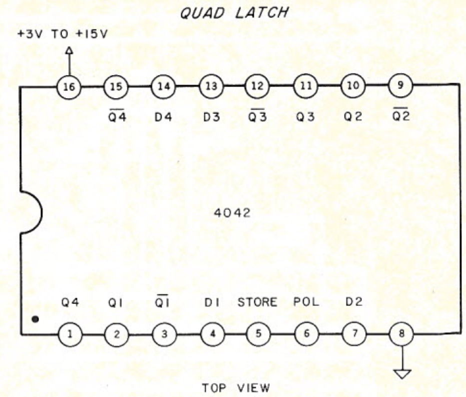


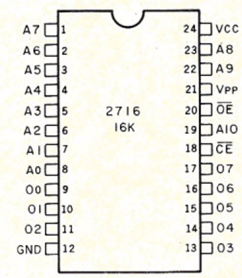
Fig. 1a. DM8131 six-bit bus comparator.



A	B	X
0	0	1
0	1	0
1	0	0
1	1	1

Fig. 1b. Exclusive-NOR gate truth table.





PIN NAMES

A0–A9	ADDRESSES
CE/PGM	CHIP ENABLE/PROGRAM
$\overline{OE}$	OUTPUT ENABLE
O0–O7	OUTPUTS

Fig. 1c. 4042 and 2716 pin configuration.

Taking 8000 hex as an example (see Table 1), bit A15 would be set high and bits A14 through A11 would be set low. When the address bus pattern matches this pattern, the output of the 8131 will go active low. Checking back to the pin-out for the 2716, pin 18 ($\overline{CE}$) has a bar over it. This means the chip is enabled when a low level is applied to the pin.

Timing

The 1802 has eight clock cycles for each machine cycle (Fig. 3, line 1). First, the high-order eight-bit byte of the 16-bit address (AD1) is available on the address bus (see line 2). Address bits A11, A12, A13, A14 and A15 are applied to the B inputs of five of the comparators in the 8131. Bits A8, A9 and A10 are applied to the quad latch (4042).

One clock cycle later, timing pulse A (TPA, see line 3) goes high. TPA is connected to the $\overline{STORE}$ (pin 5) of the quad latch (see Fig. 1c). This allows the outputs of the latch to follow the inputs; what appears at the latch's inputs also appears at its outputs.

One-half of a cycle later, the $\overline{MRD}$ (line 4) goes low; this is applied to the $\overline{STROBE}$ input of the 8131. Bits A11, A12, A13, A14 and A15 are compared to the T inputs. When the bus address matches the address programmed by the DIP switches and the pull-up resistors R1–R5, the output (pin 9) of the 8131 goes low. Since this chip enable bit will change one cycle later when the low address byte appears on the bus (see line 5 of Fig. 3), it must be latched for the entire memory cycle. To accomplish this, the chip enable bit is applied to one of the inputs of the quad latch.

Another one-half cycle later, TPA goes from high to low. This negative transition latches the outputs of the quad latch until the end of the memory cycle, when a positive transition (low to high) occurs seven cycles later. See lines 4 and 5 of Fig. 3.

The $\overline{MRD}$ is also applied to the output $\overline{OE}$ (pin 20) of the 2716. It does not matter whether the $\overline{OE}$ line is activated before or after the chip is enabled. The outputs of the 2716 are in the high-impedance state when the chip is not enabled, and therefore do not interfere with the data bus.

Now you have A8, A9, A10 and the chip enable latched in. The outputs of the latch are applied to the respective pins of the 2716. When the low-order byte (A0–A7) of the address appears on the bus, we have everything required to extract the data in the 2716 EPROM.

Now for a hint on addressing a 2732 4K EPROM. Address bit A11 has 2048 decimal located above it (Table 1). Therefore, with this bit and bits A10 through A0, 4096 bytes of memory can be addressed.

On the 2732, the programming function shares its pin with address bit A11. Subsequently, you need only to disconnect the 8131 T input from the DIP switch and the B input from the address bus for A11 and tie both high. Then connect A11 from the address bus to another 4042 quad latch and the output from that latch to the A11 pin on the 2732.

Two 2716s can also be used, but use an inverter gate instead of the quad latch. You should have enough information on addressing presented here to enable you to change the design for two 2716s. But if you get stuck, write. I will be happy to assist.

DECIMAL	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1
ADDRESS BIT	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
BIT POWERS	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
PAGE 80	1	0	0	0	0											
	DECODED BY 8131					DECODED BY 2716										

Table 1. Address bit assignment.

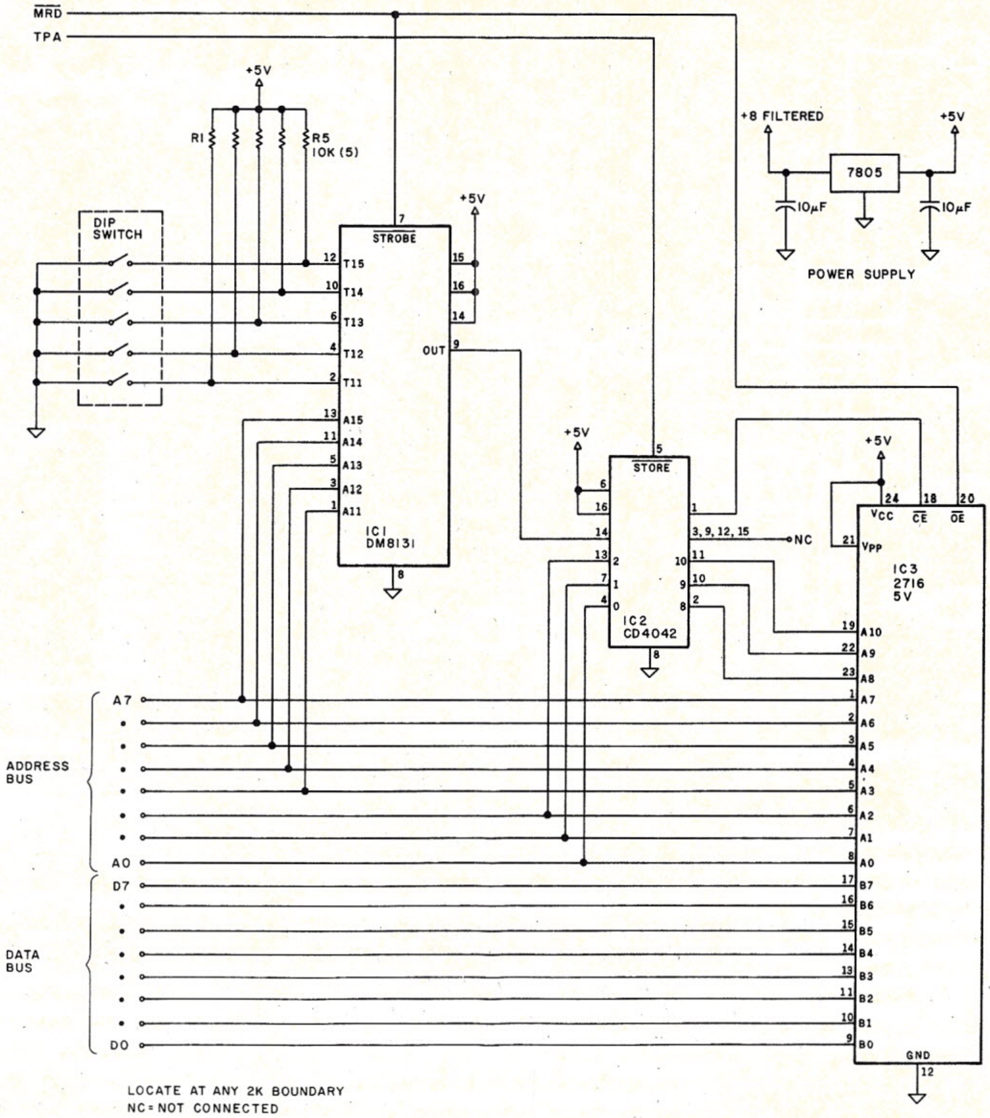


Fig. 2. 2716 EPROM circuit and power supply.

Construction

Construction can be accomplished several ways, but a single-sided printed board is probably the easiest. I recommend the positive photographic system for etching your own boards, because one of the photographic steps is eliminated. Send an SASE for a copy of the positive artwork for the PC board layout.

The 2716 can be found almost anywhere and at varying prices. Shop around for the best deal. At this writing, one source is selling them for $13. The DM8131 may be hard to find locally, so I purchased a few extra for those who have trouble. Write if you need one. Don't rush out to buy a 2732; they're asking $90. I expect them to eventually drop to around $25. Then you'll have a 4K EPROM board for your 1802 with minimum effort and expense.

Conclusion

We now have 2K bytes of 2716 EPROM with a minimum of fuss and cost. This two-IC design is probably the simplest around.

But what about programming the 2716? You can accomplish this in several ways. On one extreme you can get a friend to do it for free, or on the other extreme you can buy a $1000 EPROM programmer. More practically, many computer stores offer EPROM programming services, and a variety of homebrew programmers have appeared in the microcomputer magazines.

I am still working on the details for a two-IC EPROM programmer.

I wish to acknowledge the assistance of Gary Bergeron.

All questions and comments are welcome. Send an SASE for a reply, please.■

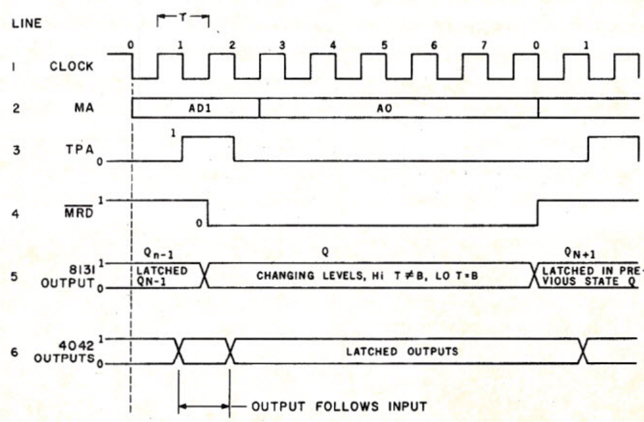
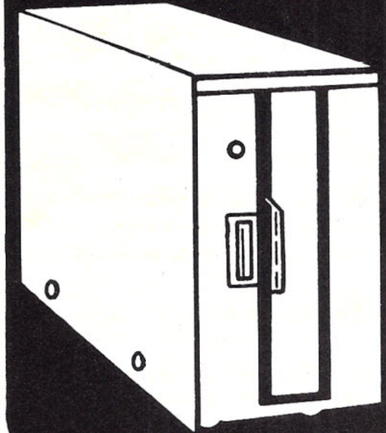


Fig. 3. Timing diagram for the 2716 EPROM circuit.

VR DATA'S DISK DRIVE HEADQUARTERS

●SALE●
TRS-80
Disk & Other
Mysteries
$19.95
reg. $22.95



77 TRACK DRIVES $545
NEW LOW PRICE
TRS-80 Model I compatible

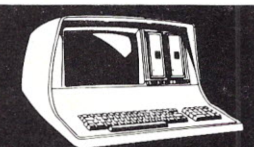
40 TRACK DRIVES $340
NEW LOW PRICE
TRS-80 Model I compatible

● ORDER NOW TOLL FREE 1 (800) 345-8102 ● IN PENNSYLVANIA (215) 461-5300 ●

HARD DISK
for TRS-80 Model II
$5995
10 meg. 5 fixed 5 removable

●SALE●
DISK HEAD CLEANERS
5-1/4" $12.95
reg. $14.95
8" 3M CLEANER
$24.95 reg. $30.00

SUPERBRAIN™
BY INTERTEC
64K $2995.00
complete with
5-1/4" disk drives ● in stock



● VR DATA Coupon ●
$5 Off!
K
ANY ITEM OVER $50.00
OFFER EXPIRES 12-1-80 ● ONE COUPON PER ITEM

TRS-80™ 64K MODEL II
$3495.00
NEW LOW PRICE

Model II Drives

1 Drive Single Enclosure	$ 899.00
1 Drive Multiple Enclosure	1069.50
Additional Drives for Mult. Enc.	540.00

SOFTWARE	MOD. I	MOD. II
Medical/Dental Patient Accounting		$1500
Word Processing (Magic Wand)		300
General Ledger	$149.95	249
Payroll	99.95	199
Data Base	149.95	299
	Tape	Disk
Upper/Lower Case Modification	$19.95	$24.95
Comprehensive Diagnostics	34.95	34.95
CP/M	$175.00	
New DOS + 40 TK	100.00	
New DOS/80	145.00	

Software Documentation Available ● CALL FOR PRICES

4K L II TRS-80	575.70
16K L II	789.60
RS-232	92.10
OK Expansion Interface	278.10
16K Expansion Interface	376.10
32K Expansion Interface	474.10
Telephone Modem	179.95
Emulator CRT by Intertec	895.00
CRT Stands from	139.00
Anti-static Mats	110.00

● VR DATA Coupon ●
$10 Off!
K
ANY ITEM OVER $150.00
OFFER EXPIRES 12-1-80 ● ONE COUPON PER ITEM

● VISIT OUR NEW WAREHOUSE SHOWROOM AND REPAIR CENTER ●

VR Data
777 HENDERSON BLVD.
FOLCROFT, PA 19032

✓64 WE SERVICE MANY BRANDS OF COMPUTER EQUIPMENT.
CALL FOR CONSULATION AND ESTIMATE.
DEALER INQUIRES INVITED ● BIDS ACCEPTED ● ABOVE PRICES ARE CASH DISCOUNTED, CALL FOR OTHER TERMS.
ORDER NOW ● TOLL FREE 1 (800) 345-8102 ● IN PENNSYLVANIA (215) 461-5300

A Video Graphics Primer

It involves more than meets your eyes.

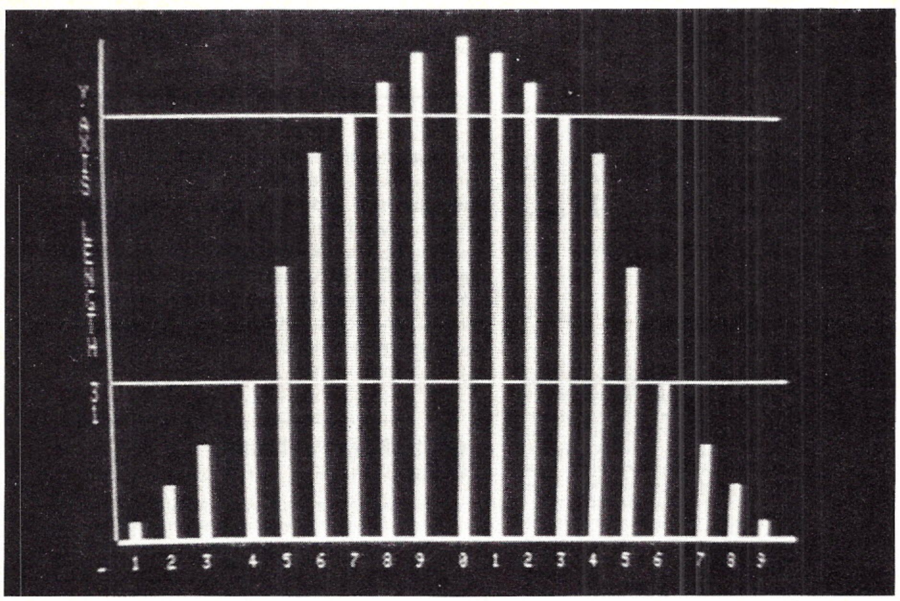


Photo 1. This representation of a normal curve is but one of the many different mathematical images possible on a home microcomputer with video graphics capability.

Jeff Knutson
1116 Morgan St.
Ft. Collins, CO 80524

If you ask the average novice or even some advanced hobbyists what video display graphics capability is, the common response may be "It's what lets the computer draw pictures on the monitor, isn't it?"

This is true but oversimplified and vague; using the computer to generate pictorial displays or, as they are also known, video graphics images is captivating.

A novice generally has no concern for the operation of the video output beyond making sure the screen is free of distortion and waviness. He isn't aware that using video graphics capabilities in a home computer depends on understanding the video output of the computer as a whole. This knowledge needn't be technical, but it does involve more than simply making sure that the VDU (video display unit) is properly adjusted.

Video Display Graphics—How to Acquire It

A hobbyist can take two routes to acquire video display graphics: buy it as a standard feature of a package system, or buy it as an expansion option.

As a rule, you can purchase a home computing system with video display graphics as a standard feature, or as an upgrade option within the unit. This is desirable; a system designed to support video graphics usually allows the user to take complete advantage of its potential. Also, many problems frequently related to programming graphic displays are reduced.

The documentation included with such systems is usually geared to helping the hobbyist. It will also usually touch on some possible applications.

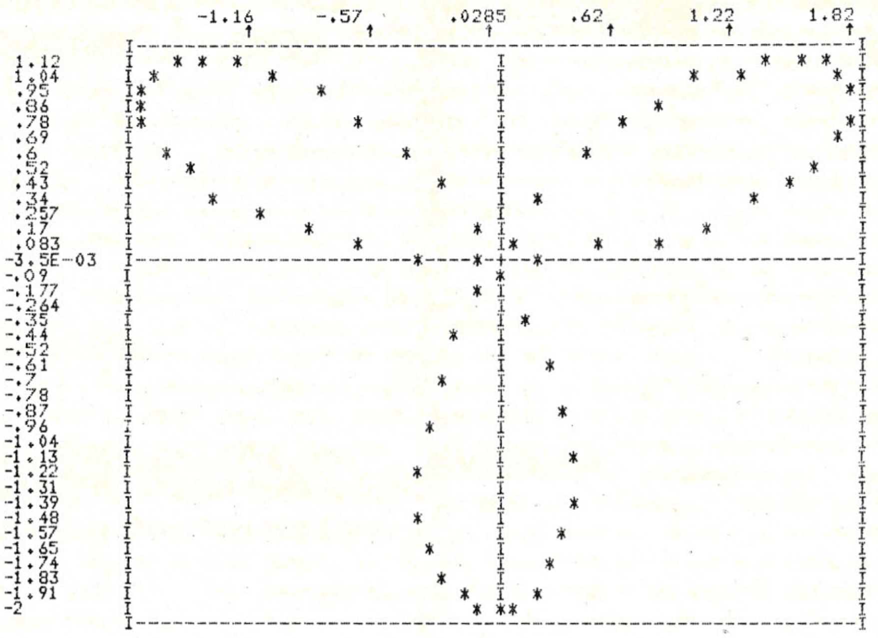
Finally, cost is the other consideration. A new system with video graphics capability as a standard feature is less expensive than buying the units separately.

On the other hand, some hobbyists may be thinking of adding video graphics to existing systems. Such a hobbyist may have bought his computer before video graphics capabilities were available, or before he could afford it.

In either case, if the money is available now, so is the capability. A variety of video expansion boards and kits is available, most for a reasonable cost. Some of the kits are designed for a certain type of terminal—the Lear-Siegler ADM-3A, for example. Others may be configured to plug into the motherboard of the computer, and operate in conjunction with the current video output circuit. Most of the boards that fall into this second area, however, are configured to fit the S-100 bus, and would be useless in any other popular bus design.

Video Graphics—What Is It?

Video graphics is the ability to generate video screen displays that convey information pictorially.



Printout. This is one example of the sort of mathematical graphics that can be executed by simply using the ASCII characters available. A program called ULTRAPLOT, from "BASIC and the Personal Computer" by Dwyer and Critchfield, generated this output.

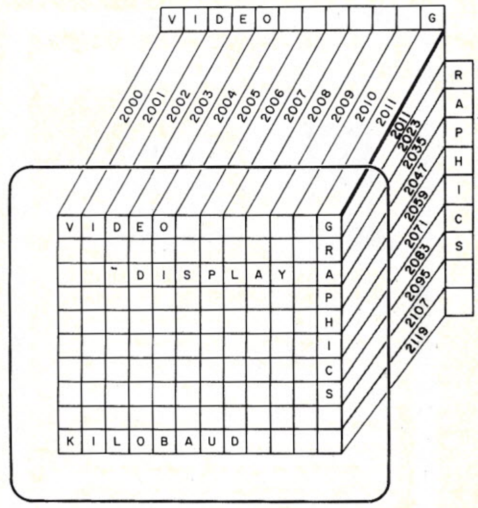


Fig. 1. Each "cell" in the video display can contain only one character at a time. Words or numbers can be made by placing the proper characters in adjacent memory locations horizontally, vertically or even diagonally.

This is different from generating alphanumeric text on the screen. For example, if the word "plane" appears on the screen, the user knows what it means. If, however, a graphic image of a plane is put on the screen, the user can comprehend the object directly.

Video graphics capability does not stop at drawing pictures. It also includes plotting data points, reproducing mathematical curves, and generating histograms and bar graphs (Photo 1).

Three forms of video graphics exist: alphanumeric graphics, memory cell graphics and bit graphics. Alphanumeric graphics can be executed on any home computer. The user develops graphic images using PRINT statements and the standard ASCII character set. One method is to use math formulas, IF...GOTO statements and PRINT commands. The other technique, if the machine dialect of BASIC will support it, is to attempt string manipulations.

In the printout, the program's output is in the form of a graph, and uses the capital letter I and hyphens (-) to draw the axes. The data points in the graph are illustrated using asterisks. The result is quite effective.

This form of video graphics is available to any hobbyist, free of charge, courtesy of his imagination. Those interested in learning more about this form of graphics capability should read *BASIC and the Personal Computer,* by Thomas Dwyer and Margot Critchfield. It contains an elementary but complete treatment of the topic. Several programs are included to help the reader explore what can be done with what is available.

This type of graphics should not be taken lightly. A talented artist can do amazing things with only letters, numerals, punctuation marks and inspiration.

The Other Two Graphics Forms—Some Background Information

The other two forms—memory cell graphics and bit graphics—require some background information before they can be discussed.

Both approaches have common origins. They employ the same basic components, and are designed with similar principles in mind. The major differences are the BASIC commands used to program images in each technique, and in how they allow the computer enthusiast to use them.

The contents of the video display are actually the contents of a given number of memory locations in the computer. The contents of these memory locations are being output directly to the video monitor. In effect, all of the changes that can be observed on the screen are the result of processor manipulation of the corresponding memory within the system.

Assume that the screen can be divided up into a series of rows of individual squares or cells. Each cell can be thought of as representing a unique memory location in the computer. Consequently, you can insert information anywhere on the screen by placing the appropriate data into the respective memory location (Fig. 1).

These cells are quite small. Each one is only large enough to contain a single discrete character: for example, the letter A. A word might be displayed by locating the letters in the correct adjacent cells on the screen.

As a result, the maximum number of letters that can be put into one row on the screen is equal to the number of cells in that row. The same is true for the number of rows of characters or lines that can occupy the screen simultaneously.

The specification that describes these dimensions of the screen is the length/line, or default, format. It is typically described as a pair of two-digit numbers, such as 80/20. The value 80 refers to the number of characters in a single line, while the 20 indicates the number of lines displayed on the screen at one time. The length/line format describes the dimensions of the video output on the screen in terms of characters.

Effective resolution is another term that comes up. The term usually refers to the smallest element of the video display on screen. It is sometimes used synonymously with the length/line format specification, but this is an error; the two are not

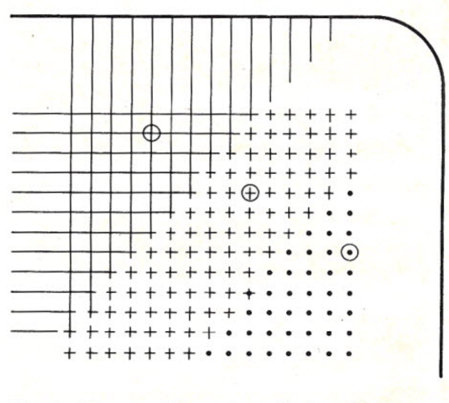


Fig. 2. The positions that pixels will appear at on the VDU are determined by the intersection points of the lines of resolution on the screen.

necessarily equal.

When the design of a video display is being evolved, the designers split the screen into a large number of horizontal and vertical lines, known as lines of resolution. More lines mean finer detail on the screen. The result is a screen full of little dots called pixels (Fig. 2). The effective resolution of a computer video display is measured by the number of pixels in its horizontal and vertical dimensions. This specification is described by a pair of three-digit numbers, such as 512/256. The number 512 indicates the number of pixels across the screen, while the number 256 describes the number of pixels down the screen.

When taken together, effective resolution and length/line determine the number of pixels in each memory cell on the screen. In essence, this determines the number of pixels under the control of a single memory location, and gives you a good idea of the degree of detail you can display on the screen at the same time.

The Main Difference between Memory Cell and Bit Graphics

The most important difference between memory cell and bit graphics is how they allow the user to control pixels on the screen.

The memory cell approach lets the user control pixels in blocks; that is, he uses the memory cells in generating graphics images. User control over individual pixels is indirect and incomplete.

The bit graphics technique places each individual pixel at the user's command. User control of the pixels is direct and complete.

This difference shows up in a variety of areas within a system, such as in language commands used to program the different graphics forms, and in the degree of programming complexity involved. How a particular manufacturer wishes to implement a particular approach in a home computer is an additional consideration. At first glance, some forms of graphics may not fall clearly into one form or the other.

In the final analysis, though, the method of controlling pixels in the display is what determines the technique used by the computer.

Memory Cell Graphics—Fill in the Squares

The memory cell graphics technique, also known as low-resolution graphics, is the less complex of the two. Each memory location has control of a discrete matrix of pixels. The size of the matrix varies widely; some of the more common sizes used are 9×7, 8×8, 5×7. This matrix is equivalent to the memory cell mentioned in the analogy. Any data placed into a memory location being output to the screen determines what will appear in that corresponding cell on the screen.

The data is actually interpreted by the video output hardware to mean "activate the appropriate pixels in cell XXXX in the video display" (Fig. 3). This is the key aspect of memory cell graphics. The user has control over the contents of any memory location, but it is the memory location that controls the pixels on the screen. Hence the

MEMORY LOCATION	MEMORY CONTENTS	REMARKS
B003	00_{HEX}	(1) MEMORY LOCATION EMPTY— —SCREEN AT THIS LOCATION IS BLANK
B003	41_{HEX}	(2) DATA IS INSERTED INTO MEMORY LOCATION
B003	41_{HEX}	(3) DATA IN MEMORY LOCATION IS DECODED BY VIDEO DISPLAY HARDWARE
B003	41_{HEX}	(4) DECODED DATA IS TRANSLATED INTO "ON/OFF" ASSIGNMENTS FOR PIXELS IN THE CELL
B003	41_{HEX}	(5) APPROPRIATE "ON" PIXELS ARE ACTIVATED ON SCREEN

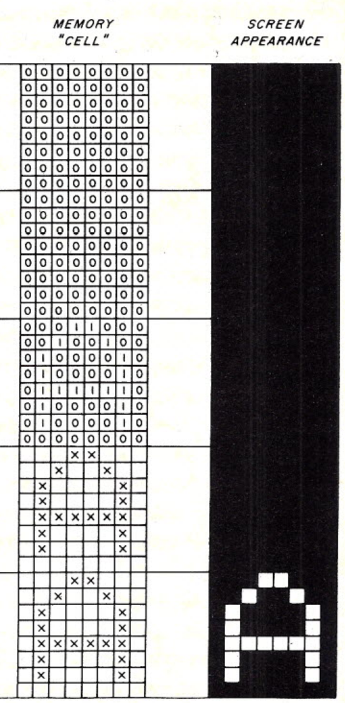


Fig. 3. The process of placing an ASCII character on the video screen might be envisioned as having the five steps shown here. In actuality, other steps are involved, but have been arbitrarily condensed for clarity.

name, memory cell graphics.

Imagine that the user wishes to place the letter A at a certain point on the VDM. He would place a certain value into the desired corresponding memory location. The video output hardware would examine the value and respond by activating the appropriate pixels within the cell to obtain a capital letter A.

If you had a light microscope, you would notice that the character would appear as a dot matrix. All of the characters that can be generated in memory cell graphics are simply values that activate various combinations of pixels in memory cells on the screen.

As you may have guessed, the entire ASCII character set the computer uses for text generation is exactly the same. The characters are graphics elements that happen to portray letters, numbers and punctuation.

Some form of memory cell output is used by most (if not all) home computers today to display alphanumeric contents on the screen. The concept of memory cell graphics is merely an extension of this idea: placing a number of abstract memory cell elements (beyond the standard ASCII set) at the user's command.

Memory cell graphics contains two types of graphics elements, and most home computers try to incorporate both. The stand-alone graphics element and the building-block graphics element both get their names from the way they are most likely to be used.

The stand-alone graphics element is meaningful to the viewer even if it is the only element on the screen. Such characters include tanks, houses and race cars. The stand-alone element is generally some form of game symbol, and is highly specific in its appearance.

The building-block element by itself isn't the least bit meaningful to a viewer. It includes many types of lines, squares, rectangular units and miscellaneous figures. The element is used almost exclusively for creating larger graphics images, and is nonspecific in appearance (Photo 2).

BASIC has two commands to program memory cell graphics: PRINT and POKE. Because graphics elements are part of the character set, just as ASCII characters are, you can display them on the screen using the PRINT command. Simply typing in PRINT CHR$ (XX) will cause the desired graphics element to print at the bottom of the screen. Like all PRINT statements, the graphics element would be scrolled upward with each succeeding line feed that occurred.

In building larger images, the POKE command is more common. POKE places the value directly into the desired memory location on the screen, and locates the graphics element without disturbing any of the previous elements on the screen. Once there, however, all memory cell graphics elements will scroll up the screen when line feeds occur.

One of the most noticeable advantages of memory cell graphics is the time it saves. A few minutes might be spent on a display that would take an hour or more on a bit graphics system. The programming of dynamic graphics offers another advantage: a minimum of memory manipulation. If I wish to orbit a ship around a planet, I can whip up a subroutine in about 20 minutes and have it running free of bugs in 20 more. The same is not true in a bit graphics system.

Also in memory cell dynamic graphics, fewer variables are moved around the screen. Thus, the equations needed to move an image around on a screen are simplified.

Finally, it is possible to easily mix text and graphics on the screen. When you label graphs and curves, or scale the axes of a display, this becomes important, as it does in many game applications.

Memory cell graphics does have a few serious drawbacks. First, the user is restricted to using the graphics element set that is programmed into the machine's ROMs when it comes. The user is not able to program his own character elements. Most manufacturers provide a large selection of graphics elements to the user, but some do not.

Also, graphics elements available for machines from one manufacturer may not be available on systems of a different make.

Finally, insufficient detail capacity is an occasional problem. The user may have a screen with XXXX number of cells on it, but once those cells are filled, no new information can be placed on the screen without losing old information somewhere. Consider, too, that most graphics elements are

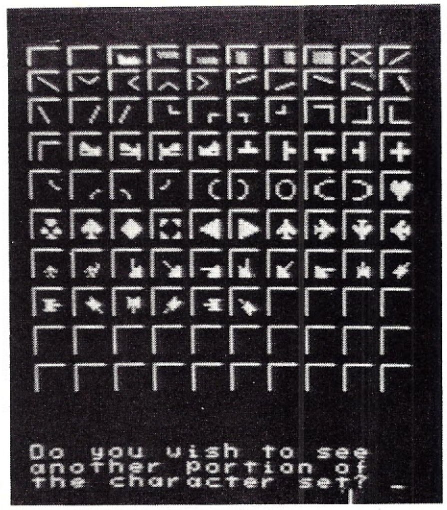


Photo 2. This excerpt of a memory cell graphics set is fairly representative of the type of graphics elements contained in most memory cell graphics systems today. The top five rows contain building-block type elements, while the lower three rows contain stand-alone type elements. Incidentally, this is also a good example of being able to mix video graphics and text on the screen.

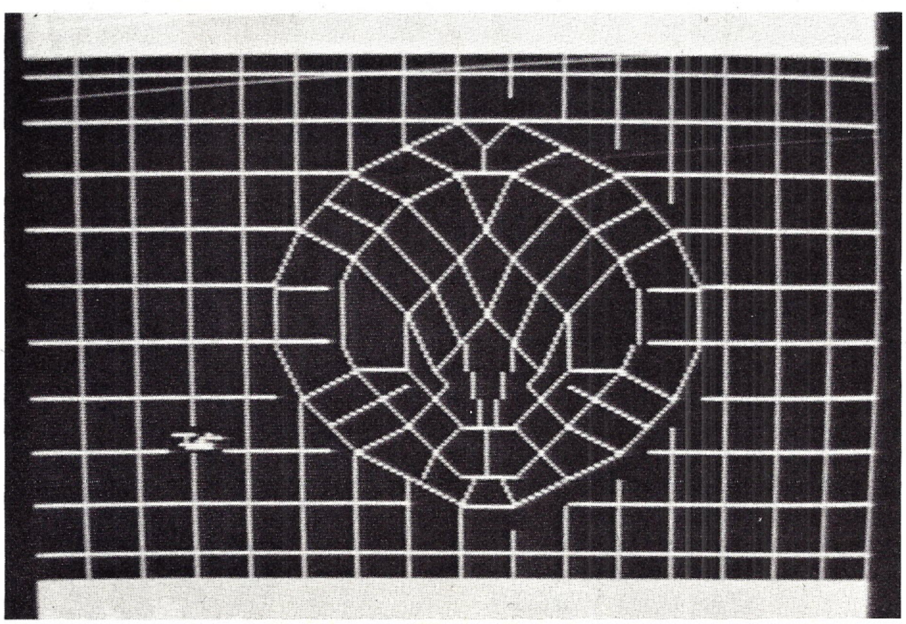


Photo 3. This graphics representation of a black hole in the fabric of space was programmed on a memory cell graphics system. The overall effect is quite convincing, but note that the grid does not connect with the black hole in the lower right-hand corner of the screen. This is a result of having exceeded the capacity for detail on this particular system.

not very detailed themselves (unless one is using stand-alone elements). So varying from computer to computer, complex images above a certain level of detail are not possible.

Because this level is not a fixed value, it usually isn't too noticeable. It proves to be a headache only when most of the screen is used. The user must simply learn to live with it.

On balance, the memory cell graphics technique is an excellent approach. The final results seldom fail to meet the programmer's expectations (Photo 3).

Bit Graphics

With bit graphics—or, as some purists prefer to call it, "true" graphics—the user has control over every individual pixel in the display by discarding the length/line format used in text generation. Instead, the computer uses a direct pixel by pixel on/off control of the screen.

This does not change the fact that some form of memory is still being output to the screen. The difference is in how the video display hardware decodes the values in the respective memory locations.

In memory cell graphics, a single memory location controls the contents of a memory cell by turning the individual pixels within the cell on or off to generate the appropriate element.

In bit graphics, the different bits of the byte in the memory location control the individual pixels assigned to that memory location (Fig. 4). This is the key aspect of bit graphics. The user has control over each individual pixel on the screen by manipulating the necessary bits in the corresponding memory locations.

Bit graphics can be used to plot points or to draw a larger overall image.

The point-plotting function is unique in that the user can address the pixel on the screen with a pair of Cartesian coordinates. To plot a series of points, the user types the coordinates into the computer. It responds by activating the pixels addressed.

The plotting function can be tied into programs too, with the result that complex mathematical curves can be duplicated with a high degree of precision (Photo 4).

A new capability associated with bit graphics, vector graphics, can greatly simplify generating graphics images. When a pair of points has been plotted on the screen, the user may instruct the computer to connect them. The computer will respond by "drawing" a line between the points specified.

Liberal use of vector graphics is effective in creating an overall detailed image. Even so, programming an image in bit graphics can be time-consuming.

The POKE command, and usually some form of a PLOT command, are the BASIC commands used to program bit graphics. The PRINT command cannot be used here because pixels are not part of the character set.

The PLOT command informs the video output hardware that a certain point is to be placed on the screen, and that the address of the point will be a pair of coordinates, not a direct memory address. The function is easy to use, too—the command is entered as PLOT XX,YY, and the point is plotted.

In bit graphics, the POKE command serves as a software on-off switch. In poking a memory location, the user is only inserting the desired bits into the location. In units which can implement a variety of color tones on a color monitor, the POKE command also may be used to select the desired color of pixels.

Bit graphics systems have some sizable advantages. The greatest one is also the most obvious—finely detailed images (Photo 5). The bit graphics user does not have to contend with a restrictive graphics element set.

This also allows another choice—the scale of the image that the user wishes to portray. The same object can be programmed big or little. The scale of an image that can be programmed in a memory cell graphics system is much more limited.

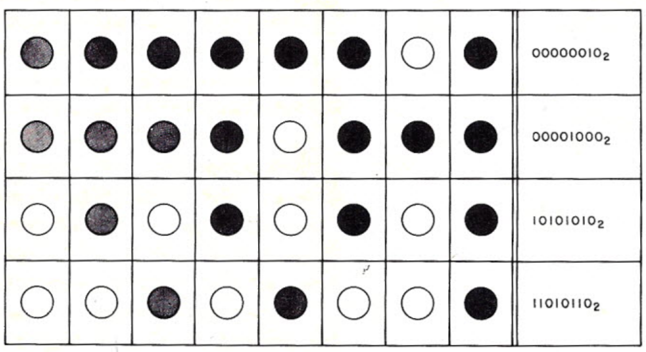


Fig. 4. Bit by bit control of the pixels on the video screen has been implemented in a wide variety of ways by different manufacturers. Most techniques in use in home computers using bit graphics are far more complex than the method shown here.

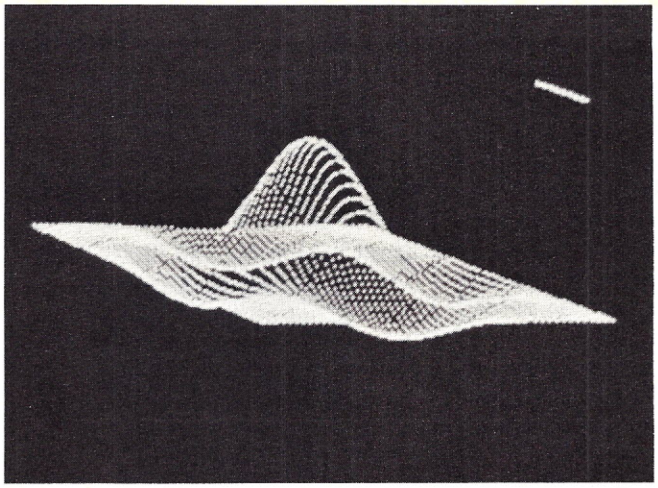


Photo 4. Here, a sync function has been plotted on a bit graphics system. Because bit graphics is set up to perform true point plotting, it is well suited to reproducing complex mathematical functions. Programming this display, however, was not a quick effort, according to the programmer.

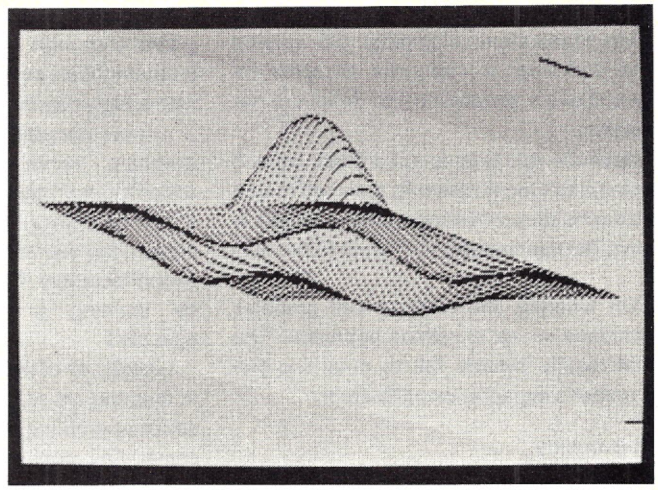


Photo 4a. This is the same sync function as before, but shown in inverse format. Both memory cell graphics and bit graphics can support the inverse function, and both can support other graphics enhancing features as well. These include underlining, blinking and inverse blinking video output.

Bit graphics also allows true point plotting on the screen. Analogous functions can be programmed for memory cell systems, but they are all subject to the flaws of being an imitation.

Bit graphics has its faults. Unfortunately, there is no such thing as a small image. Because a minimum number of pixels is always needed, only large images and larger images exist. Thus, even though some computers have software to minimize the problem, programming takes time.

The problem lies not only with having to turn on all of the pixels needed for a display. Discovering which memory locations control which pixels is difficult. The memory mapping of the video display is different from memory cell graphics, and is much more complex.

Programming dynamic graphics is also more complex in bit graphics than it is in memory cell graphics. The motion of hundreds of pixels, rather than a few memory locations, has to be programmed. Experience, however, usually takes care of any problems.

Bit graphics' extreme memory consumption is one problem that can't be solved. Each pixel on the display is being controlled by a unique bit in a given memory location, and a bit graphics system with a high effective resolution needs a large amount of memory to support itself. Memory is expensive—the more that is needed by the system, the more the system is going to cost.

Also, the bit graphics system makes it difficult—though not impossible—to mix text and graphics on the screen. One solution is to set up a symbol table for the text when writing the program. In essence, the programmer is designing his own set of character elements. The other solution is to buy the appropriate hardware for the system.

On the whole, bit graphics can be challenging to learn to program. But most of the difficulties are related to software complexity, and given time any user will be able to surmount them.

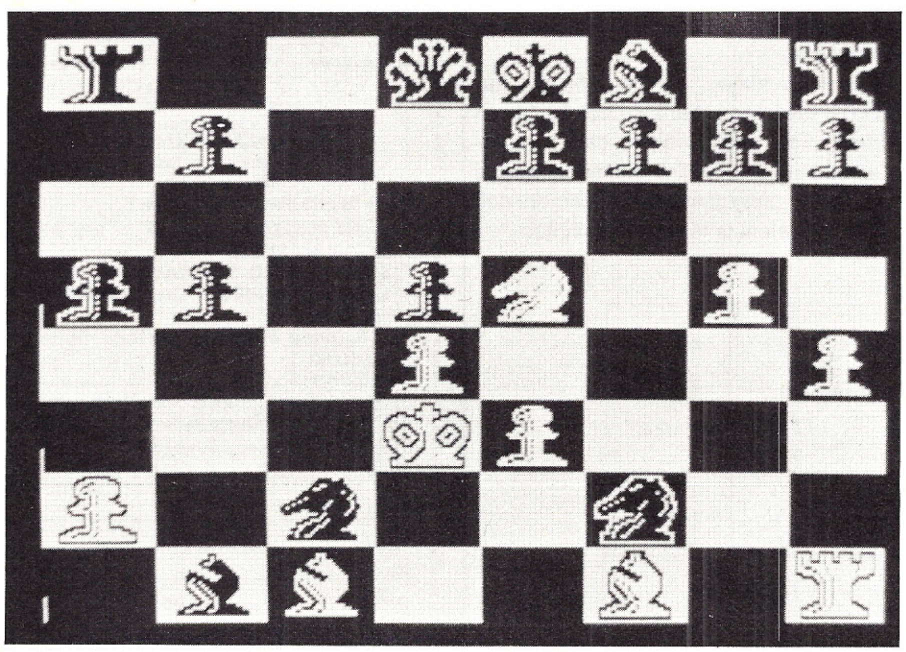


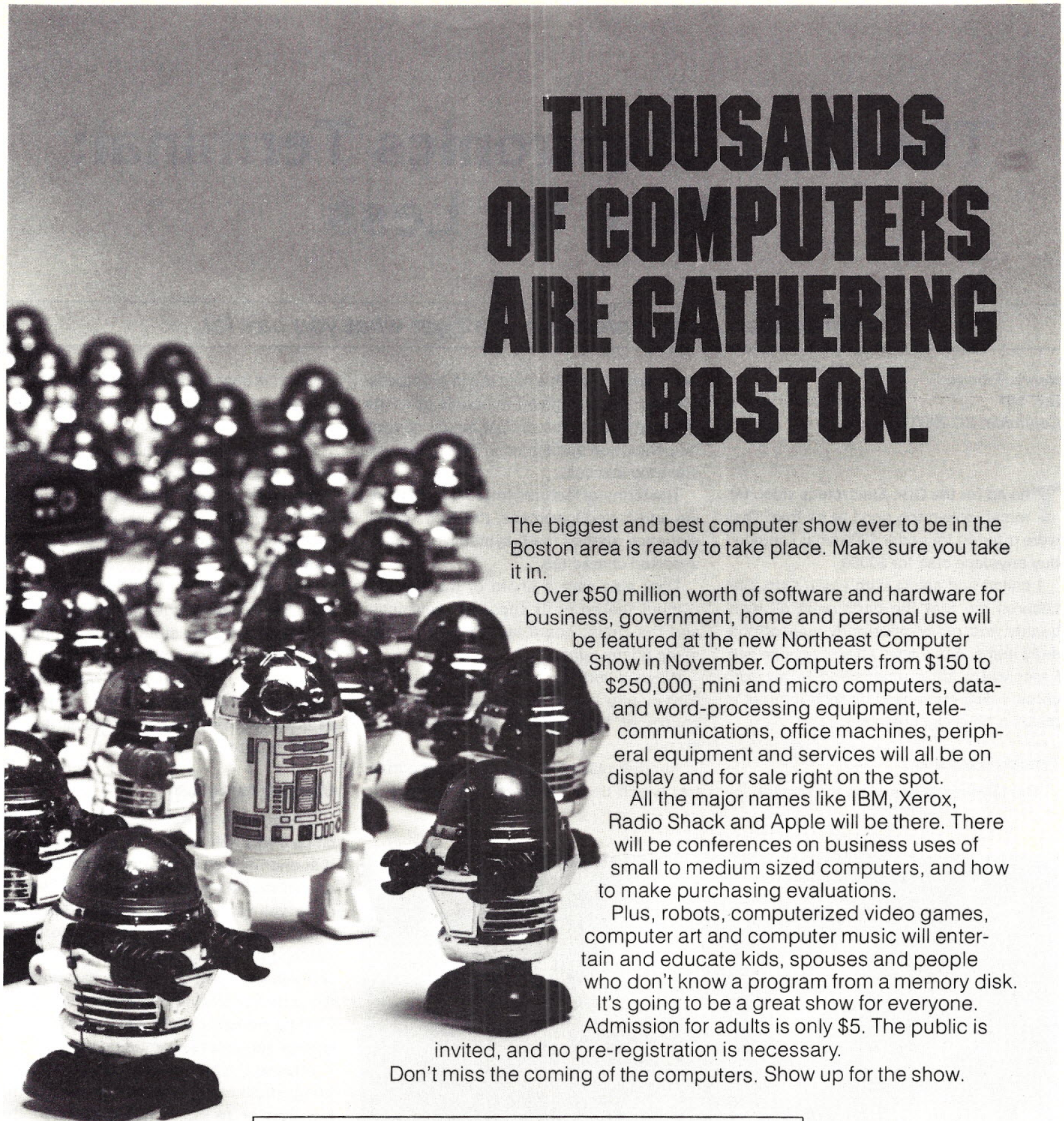
Photo 5. This is a good illustration of the degree of detail a bit graphics system can support. This display is part of the chess program Sargon. A large portion of the memory consumed by Sargon is for graphics manipulation only.

Some Closing Thoughts

Hobbyists considering acquiring video graphics capability should keep a few considerations in mind. These points will help ensure hobbyist satisfaction with the approach he chooses.

First, neither form of video graphics is better than the other. Each approach has its strong and weak points. Understanding the capabilities of each approach is the most important part of the hardware.

Next, the hobbyist is a large part of the equation. He should take a good look at where his interests lie, and how much time and effort is needed to use a specific graphics form. It would be a shame if a part-time enthusiast bought a bit graphics system, but never used it for lack of time. It would be just as bad to have a real aficionado buy a memory cell system, only to discover it can't meet all of his expectations. ■



✓ Reader Service index—page 241

The Otto Electronics Terminal: More for Less

Fortunately, you sometimes don't get what you pay for.

Henry Roberts
U.T. 161
Columbia, SC 29201

The ad for the Otto Electronics video terminal looked too good to be true. They were offering for under $300 what I couldn't buy anyplace else for $2000.

I called and spoke with Linda Otto. She assured me that the parts were all high-quality, and, more importantly, were off-the-shelf items. A few days after my phone call, I received a letter responding to my concerns. I was impressed by their interest in me as a potential customer.

Terminal Features

The terminal is based on the Mostek terminal controller. This chip is a microprocessor and is programmed to handle all of the terminal's activities except actual character generation and memory. This helped explain the low cost.

The terminal features full cursor addressing, upper and lowercase, the entire Greek alphabet, special math symbols and other assorted characters.

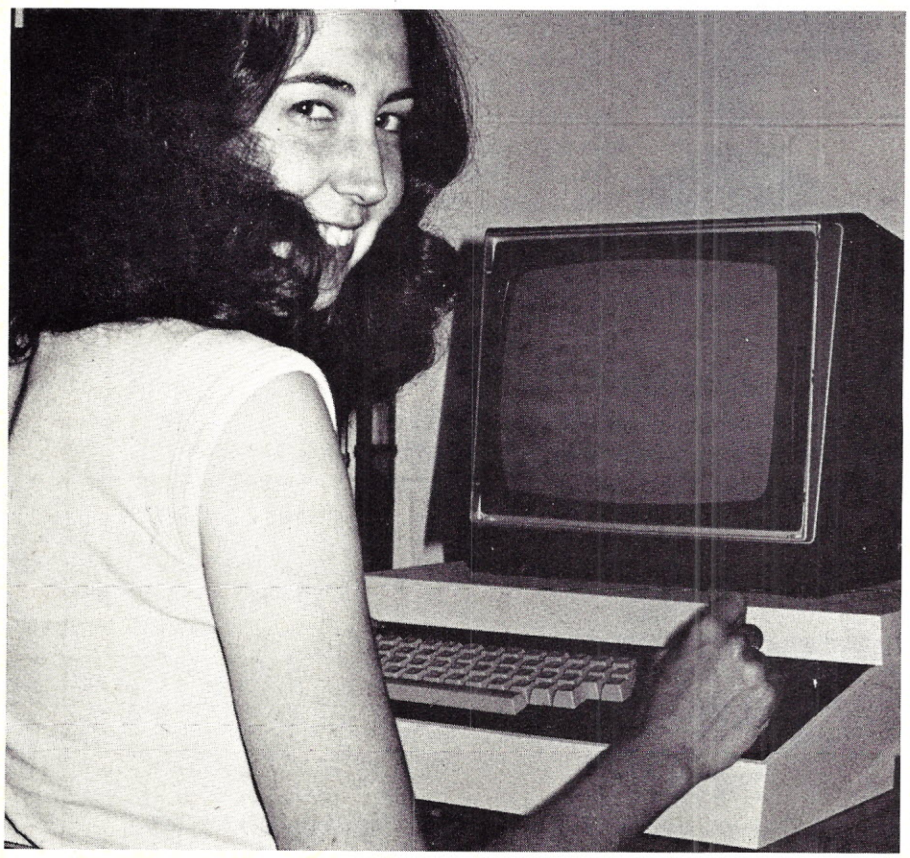
They also sent a photo of the terminal without the cover. It appeared well-made, and the circuit board looked simple. That made up my mind.

When it arrived I was surprised at the high-quality parts. For example, all but a couple of the ICs were major American brands.

Another pleasant surprise was the manual, which uses the Heath manuals as its guide. It is simple enough for anyone to follow, has a troubleshooting section and a complete operating guide.

The kit, which I assembled in about nine hours, was not difficult. The point-to-point wiring was simpler than with most other kits I've wired. (The kit assumes that you know how to solder and have access to a voltmeter.)

The only problem I had was some missing parts in the George Risk keyboard. The parts were two resistors and a capacitor. They were easy to replace, and later I discovered they had no function. Still, it would have been more convenient if there had been a note advising me that the parts were missing, but not needed.



Checking the System

Since the unit had sockets for all the chips, I was able to check the power supply first and thus avoid blowing all the chips at once if the voltage polarity were backwards or the voltage itself were too high. There are three separate voltages in the Otto terminal, and all but one were in tolerance. The +5 volt line read zero. I quickly switched the terminal off and measured for a short. Finding none, I checked the inputs to the voltage regulators, everything was OK.

I traced the problem to the output of the bridge rectifier that converts the ac from the transformer to dc. A quick check of the bridge rectifier with an ohmmeter showed that it was all right. That puzzled me for a minute. The PC board was beautiful with heavy plating, but I took my meter and checked the connections on the board between the bridge and the voltage regulators. They were OK.

It turned out that one of the holes for the rectifier itself wasn't plated through. The output from the rectifier fed a trace on the top of the board, and, of course, I had soldered it on the bottom. Fixing this proved difficult since the part covered the pad on which it rested.

I put plenty of solder on that lead of the bridge and plenty of solder on the top pad that wasn't plated through, filling the hole.

Holding the rectifier in place, I applied heat directly on the tinned lead of the bridge and, when the solder melted on the top of the pad, quickly pushed the bridge flat on the board. Then I turned the board over and applied solder to all of the unsoldered pads. A quick check with my ohmmeter showed I had fixed the problem.

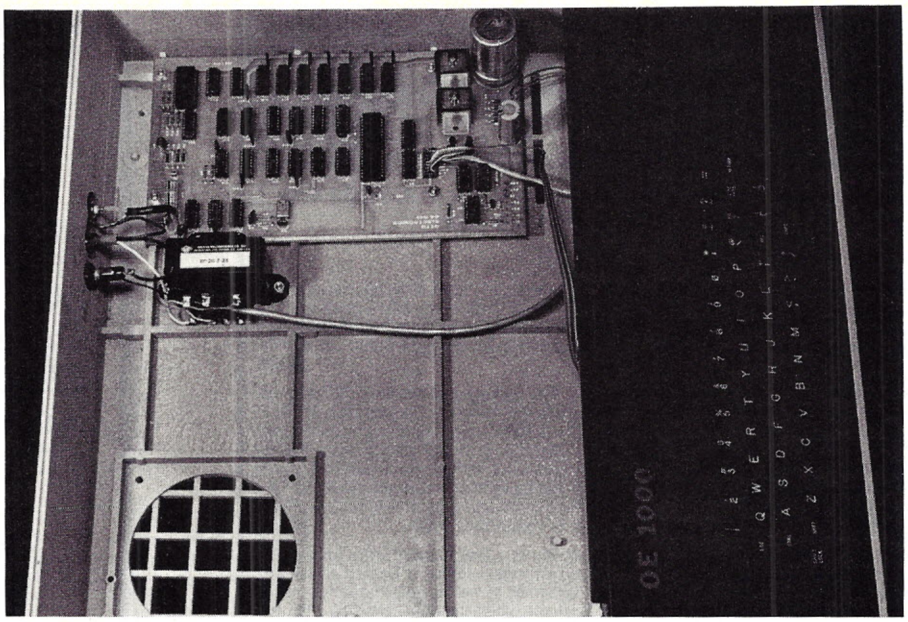
I grounded myself to a cold water pipe (taking care to stay away from everything electric) and installed the 33 chips. After quickly plugging everything together, I flipped the power switch and waited. Nothing happened.

Using an oscilloscope, I found that signals weren't passing through the shift register that converted the parallel output from the Motorola character generator to serial video.

I called Neil Otto and told him about the defective part. He said he would send a replacement by first-class mail the next day. The problem, he explained, was due to TI back-ordering that particular part. Neil had bought a lesser brand to fill his back orders.

He told me it was the last time he would ever do that, because he had ended up replacing a lot of the parts. Buying from people such as TI was his only assurance of good parts, since he didn't have the facilities to test chips in quantity.

With the new parts, everything worked instantly. I sat and played for a while and enjoyed the upper and lowercase alphabet. I tried entering the control characters for the Greek alphabet. My wife, who is a mathematician, wanted to see all the math symbols, such as square root and integral signs. I waited to connect it to the computer while the family, including our two children, played with the terminal for the rest of the day.



Inside the Otto Electronics video terminal.

Connecting It to the Computer

The next evening I hooked the terminal up to the serial interface of my computer. Everything worked perfectly.

One of my main concerns when I bought the terminal was its connection to the Southwest Technical cassette interface, which is in series with the computer. Southwest Tech's literature repeatedly warns that you must have access to your terminal's 16X baud rate clock or buy a separate serial interface. Also, the software-controlled features of the cassette interface require decoding circuits in the terminal.

But the 16X baud rate clock only provides a 4800 Hz tone. This tone is divided down, in the cassette interface, to provide the 2400 and 1200 Hz tones used in the Kansas City Standard recordings. This means you must have a 4800 Hz signal to record directly from the terminal. I solved the problem by connecting the computer's 16X baud rate clock to the terminal's 16X baud rate clock with a jumper along the back of the PC board in the cassette interface. Do not run the jumper underneath the board, since this causes cross-talk, and you will have difficulty reading binary tapes.

Using the automatic functions of the cassette interface proved to be almost as easy. Appendix A of the *Southwest Tech Assembler Manual* gives a list of four connections from the serial interface of the computer to the control input lines on the cassette interface.

These connections are between the LSI chip on the serial interface of the computer and the center edge connecter along the rear of the cassette interface PC board. I am considering pin 2 as ground on the AC-30 cassette interface (see Table 1).

A Year Later

The terminal has worked well. The extra characters and the addressable cursor have added a new dimension to my programming. I have had no failures of any kind with the terminal. Keyboards are one of the first things to give problems in equipment such as this, but I have had no keybounce or failure of keys to enter properly. The keys feel right, and this is important to me as a touch typist.

Problems

I did find a few relatively minor problems. Control-C does not stop endless printing loops very well. This has always been a problem with Southwest Tech computers, but is worse with the Otto Electronics terminal. The problem is caused by the fact that parallel-to-serial conversion is done by software at both ends. Southwest Tech uses MIKBUG, which adapts a parallel interface to serial. Switching to SWTBUG and a serial control interface, I am told, will stop BASIC cold with one control-C.

I could not change the control-H to a DEL in Southwest BASIC. The control-H will back-space, but I would rather use the delete key since it also erases the characters. The problem, I discovered, is with Southwest Tech BASIC: It will not accept 7F for any purpose, even in a literal print statement.

Conclusions

This video terminal is a fine piece of equipment and compares well with terminals costing several times the price. If you are in the market for a high-quality video terminal, I strongly suggest that you take a close look at the Otto Electronics terminal. The extra features alone make it worthwhile. It is available from Otto Electronics, PO Box 3066, Princeton, NJ 08540.

The only disadvantage I can find is that it is fixed at a 300 baud rate. This is not a problem for me since I don't own a printer and could not read a faster baud rate as the lines scroll by.

The price tag was my original reason for purchasing the Otto Electronics terminal. It proves the exception to the rule, "you get what you pay for."■

From	To
MP-C IC1 pin 7 (read on)	AC-30 pin 9
MP-C IC1 pin 4 (punch on)	AC-30 pin 10
MP-C IC1 pin 6 (read off)	AC-30 pin 7
MP-C IC1 pin 5 (punch off)	AC-30 pin 5
MP-C IC1 pin 1 (ground)	AC-30 pin 2

Table 1.

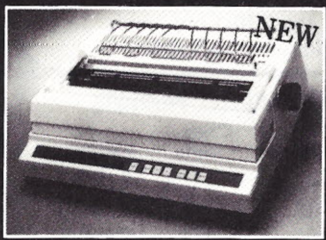
Data Terminals From MICROMAIL? YES,

Because We Offer....

...A 'Personal Approach'
Towards the Quick and Efficient Handling of Your Individual Order.

...Inventory.
The Equipment You Select is Readily Available from Our Stock.

...Terminals Only.
We Specialize in Data Terminal Equipment.



DIABLO 630

- Uses metalized or plastic print wheels.
- Automatic bi-directional printing.
- Maximum print speed is 40 characters/second.
- Variable column spacing, 120 positions/inch.
- Variable line spacing, 48 positions/inch.
- Forms control; bi-direcitonal paper feed; horizontal and vertical tabs; left, right, top and bottom margins.

$1999.00
Optional Forms Tractor — **$200.00**

DIABLO 1650

- Prints at 40 cps, using 88, 92, or 96 char. metalized printwheels.
- Vertical resolution 1/48". Horizontal 1/120". Capable of proportional spacing, bidirectional printing, and graphics under software control.
- Bidirectional normal and direct tabs. Left, right, top and bottom margins.

R.O. $2890.00
KSR $3155.00

DIABLO 1640

- Uses plastic printwheel and prints at 45 cps. Otherwise, shares identical features with 1650 including:
  - Friction or tractor feed, up to 15" wide.
  - Cartridge ribbon, fabric or carbon.

R.O. $2745.00
KSR $3050.00

T.I. 810

- Includes upper/lower case option.
- Bidirectional printing at 150 cps.
- Tractor-feed forms, 3" to 15" wide.

$1599.00

Options:
- Forms length control — $100.00
- Vertical Format Control with Compressed Print — $125.00

DECwriter LA 34

(Shown with optional forms tractor and numeric keypad).

- Prints 10, 12, 13.2, or 16.5 characters per inch, upper/lower case.
- 2, 3, 4, 6, 8, or 12 lines per inch.
- Friction feed, paper width to 15 inches.

$969.00

Options:
- Numeric keypad — $80.00
- Adjustable forms tractor — $130.00

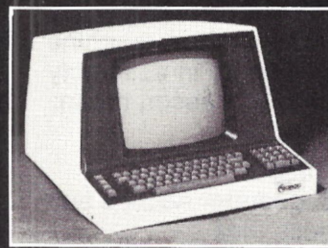
NEW

Model 'AA' $1,099.00

ANADEX DP-9500/9501

- High Density Graphics
- Parallel, RS-232C, and Current Loop interfaces standard.
- Double width printing
- 132/175 or 132/220 columns.
- 50 to 220+ lines/min., 150/200 CPS 9 x 7/7 x 9 font or 120/200 CPS with 11 x 9/7 x 9 font.
- 9-wire print head, 650 million character life.
- Bi-Directional printing with shortest distance sensing logic.
- Adjustable width tractor paper feed.

Call For Low Price



SOROC IQ 120

- Displays 80 x 24, upper/lower case.
- Separate numeric keypad and cursor keys.
- Protected fields displayed at reduced intensity.

$689.00

SOROC IQ 140

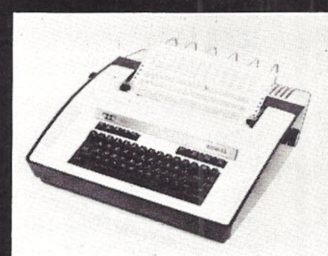
- 117-key detachable keyboard with numeric cluster and cursor control.
- Insert/delete line, insert/delete character.
- Underline, blink, reverse, ½ intensity, protected and blank fields.
- Printer port with independent baud rate — prints line, partial or full screen.

$1099.00

TELEVIDEO 912/920

- Insert/delete line, insert/delete character, line/page erase.
- Reverse video, blinking, underline, ½ intensity, protected field, blank security field.
- Uses 7 x 10 dot matrix for a high quality u/l case display with descenders.
- Standard typewriter or teletype keyboard; numeric keypad.
- Model 920 includes 17 dedicated keys for function and editing.
- Block or character transmission, auxiliary printer port.
- Cursor up, down, left, right return, home, load, read, tab and back tab.

Call For Low Price



TELETYPE 43

- Prints 132 columns, upper/lower case with true descenders.
- 30 character/second print speed. 110-300 baud.
- Uses 12" wide by 8.5" pinfeed paper.
- Print position scale, paper guide and supply rack.

$999.00

MICROMAIL ™

MICROMAIL • BOX 3297 • SANTA ANA, CA 92703
(714) 731-4338

Write or Call In for Our Free Catalogue!

TO ORDER: Send check or money order to: **MICROMAIL, P.O. Box 3297, Santa Ana, CA 92703.** Personal or company checks require two weeks to clear. All equipment includes factory warranty.

SHIPPING: We ship freight collect by UPS when possible. Larger terminals are shipped by motor freight. Air and express delivery is available on all products.

HANDLING: All orders are subject to MICROMAIL's handling charges. Less than $750.00, add 3%. $750.00 to $2,000.00, add 2%. Over $2,000.00, add 1%.

Microcomputer Hardware For the Handicapped

Single-key data entry for the PET.

Alfred J. Bruey
201 S. Grinnell St.
Jackson, MI 49203

Not everyone is able to use a keyboard to enter data into a microcomputer. A severely handicapped person may have to rely on a headswitch, kneeswitch or footswitch. In such a case, a switch to be used by whatever muscle that person can control with accuracy must be designed.

The switch hardware and software described here were developed as part of a test project to develop a scanner-type communication device for severely handicapped people. The project differs from others because this one has been done in BASIC for the Commodore PET, with the switch input implemented on the user port.

The alphabet, the digits from 0 to 9 and a few special symbols are displayed in four rows on the bottom half of the PET screen. A cursor moves down the left side of the screen, stopping for one second at the end of each of the four rows.

When the cursor stops at the end of the row that contains the character you wish to display, you press a button. Then the cursor moves across that particular row, stopping for one second under each character. When it stops under the desired character, you press the button again. The selected character is displayed on the top half of the PET screen.

By repeating this process, you can build a message on the top half of the screen. You can also select special symbols to play a note, to erase a character, to erase the entire message or to return to the main menu.

The Hardware

The only parts you need are an edge connector for the user port (available from AB Computers, 115 E. Stump Rd., Montgomeryville, PA 18936), some wire, a 10,000 ohm (10k) resistor and a switch (a momentary, normally open, push-button switch is best). Three connections must be made to the edge connector, to pins GND, PA0 and PA1 (Fig. 1).

The circuit is shown in Fig. 2. Make the wires from the switch to the edge connector long enough to use from in front of the PET.

The Software

The edge connector points that you just made connections to are from the PET's 6522 PIA. You need two memory addresses for this application—location 59459 is the direction register and location 59471 contains the values of bits PA0 to PA7.

Assume we enter the command POKE 59459,7. This will make pins PA7, PA6, PA5, PA4 and PA0 input pins and PA3, PA2 and PA1 all output pins. This is because 7 decimal is 00001110 binary; the 0 in a position makes that position an input port and the 1 makes it an output port. Now you can use a POKE to location 59471 to send signals to the output pins and a PEEK at location 59471 to see if any information has been placed at the input pins.

For our example, we want to make PA0 an output pin and make the rest input pins. The command

```
POKE 59459, 1
```

will do this. Next we want to set the output pin high (to a 1). For this we use

```
POKE 59471, 1
```

Now pin PA0 will stay high throughout the run.

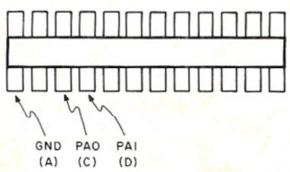


Fig. 1.

Let's look at what happens in Fig. 2 when the switch is both open and closed. When the switch is open, pin PA0 will be high because we set it that way. Pins PA7 to PA1 will be high because PET 6522 PA0–PA7 pins are high if they are input pins and not connected to anything.

Therefore, if we want to tell if the switch is open, we PEEK at location 59471. Since all pins are high, we should find a decimal 255, since this is 11111111 binary.

If we close the switch, PA1 will be the only pin to change. Since closing the switch connects PA1 to ground, it will go to 0. Thus, peeking at location 59471 will return 253, which is 11111101 binary.

Now enter the following short BASIC program. It doesn't do much, but it shows you how the switch works.

```
10 POKE 59459,1
20 POKE 59471,1
30 Print PEEK(59471);
40 GOT 30
```

Run this program. You should see 255s on the screen when you are not closing the switch, and 253 when you are closing it. Try to hold down the switch just long enough to get one 253.

It's going to be hard, especially if you have a bouncy switch. This problem has two solutions—one hardware and one software. You could design hardware that would allow a switch closing to be captured just once, ignoring the extra length of time that you allow the button to be pressed. The software method is to sit and idle in the program until the button has been released.

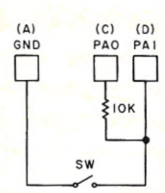


Fig. 2.

The following program does this.

```
10 POKE 59459, 1
20 POKE 59471, 1
30 IF PEEK(59471)=255 THEN 30
40 PRINT PEEK(59471);
50 IF PEEK(59471)=253 THEN 50
60 GOTO 30
```

This program should print a 253 every time you press your button to close a switch. Using the ideas in this example, you should be able to incorporate a switch into your own programs.

Extensions

You might want to try several extensions:

Use a photoresistor circuit as the switch. Your PET could be controlled by turning a light on and off.

You can put more than one switch at a time on this port. By decoding location 59471, you can tell which switches are closed. ■

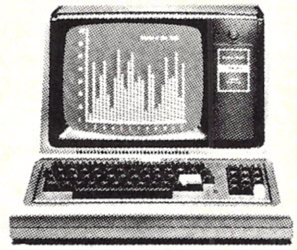
MODEL II

26-4002
64K 1 Drive
$3499.00

AUTHORIZED
TRS-80® DEALER A301

COMPUTER SPECIALISTS

MODEL I



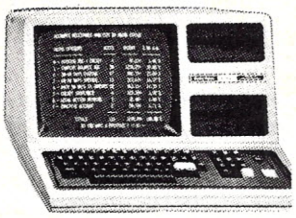
26-1054
4K Level II
$552.00

26-1056 16K Level II System with Keypad. $670.00

26-1145 RS-232 Board	84.00
26-1140 "O" K Interface	249.00
26-1141 "16" K Interface	365.00
26-1142 "32" K Interface	476.00
26-1160 Mini Disk - Drive O	424.00
26-1161 Mini Disk - Additional	424.00
26-1154 Lineprinter II	720.00
26-1156 Lineprinter III	1799.00
26-1180 Voice Synthesiser	339.00
26-1181 VOXBOX	145.00
26-1104 Factory Upper/Lower Case Modifcation Installed	70.00
26-1506 Scripsit - Tape	60.00
26-1563 Scripsit - Disk	85.00

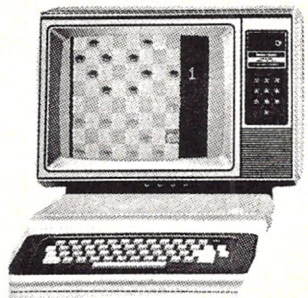
NOTE: Call for availability of VIDEO TEX, Model III, Color, and other new products.

MODEL III

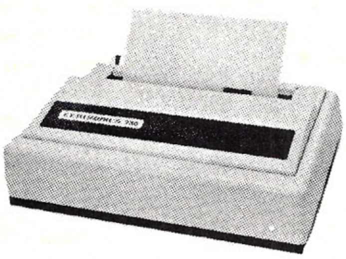


26-1061 4K I. $630.00
26-1062 16K III. 900.00
26-1063 32K III
2-Drives, RS232. . . . . 2246.00

COLOR



26-3001 4K. $360.00
26-3002 16K. 540.00
26-3010 Color Video. 360.00
26-1206 Recorder. 54.00
26-3008 Joysticks. 22.50



CENTRONICS

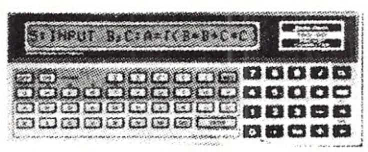
Fast 100 CPS Centronics
730 Printer. $675.00
Text Quality Centronics
737 Printer. $850.00

ALL OTHER R.S. SOFTWARE FURNITURE, STANDS, CABLES AND ACCESSORIES DEDUCT 10% FROM CATALOG PRICE

Novation Cat Modem. . $149.00
CCA Data Management System. 72.00
Adventure Games
Games 1-9 each. 14.00

Model II Cobol Compiler
$360.00
Cobol Run Time Package
$36.00

Pocket Computer



26-3501 1.9K P.C. $225.00
26-3503 Cassette I/F. 45.00
14-812 Recorder. 72.00

Acorn Software Products, Inc.

GAMES:
Alien Invasion. $9.00
Stock Market. 9.00
Star Trek. 9.00
Block 'Em. 9.00
Ting-Tong 9.00
UTILITIES:
System Savers. 14.00
EDUCATION:
Language Teacher. 18.00

FREE: PRICE LIST UPON REQUEST

1-800-841-0860 Toll Free Order Entry

✓100

MICRO MANAGEMENT SYSTEMS, INC.

No Taxes on Out Of State Shipments

Immediate Shipment From Stock on Most Items

DOWNTOWN PLAZA SHOPPING CENTER
115 C SECOND AVE. S.W.
CAIRO, GEORGIA 31728
(912) 377-7120 Ga. Phone No.

Full Factory Warranty on All Items Sold.

Largest Inventory In the S.E. U.S.A.

*TRS-80 is a registered trademark of the Tandy Corp.

So I Bought This Computer

How a microbusiness uses an Apple II for fun and profit.

David C. Goodfellow
13026 13th S.W.
Seattle, WA 98146

When my microbusiness—writing, editing, illustrating and page layout for technical manuals, sales brochures and whatever else my customers want in the printed word—started to take off a couple of years ago, my production became increasingly limited by my old Itel paper-tape word processor. The speed wasn't too bad, but the Itel's Selectric typewriter was beginning to shake itself apart, generating errors in embarrassing places, such as final copy. This slowed me down considerably, for I had to watch while it typed, and correct errors as they occurred.

So in June 1978 I started looking at alternatives. The IBM Electronic Composer was too expensive, and its memory was too limited. IBM's MTSC was also too expensive. I was about to buy a used CPT word processor when I wandered into a microcomputer store and was bitten by the bug.

Back then, you couldn't find a micro to do quality word processing. But I was sure that it would soon be available, probably by the time I'd learned how to use the computer. And while I waited for word processing, the computer could keep the books, print the bills, write the checks, clean out the office and play games.

An Apple II

The day I came home with my Apple II (16K, with one disk drive), I found two new jobs waiting. So I set the computer up in the living room and gave my family carte blanche. After showing them where the manual says that the only way you could hurt the Apple was to type on it with a hammer, I disappeared into my basement office. When I finally emerged two months later, I found my 12-year-old daughter was an expert programmer, and she'd half worn out the electrons.

I set up my Apple next to my worn-out Itel word processor (which I was destined to use for another year). Its very presence started bringing in new customers. It didn't matter that I didn't know how to use it. The Apple brought in $15,000 worth of business in 1979 that I otherwise wouldn't have had.

Naturally, the Itel couldn't hack it. My work week increased from about 45 hours to nearly 70. While the Itel clanked and groaned in harmony with my aching back, the Apple sat there and smirked.

I devised a plan whereby I could learn how to actually use the computer—peeks, pokes, calls, the whole bit—thus making my work easier through understanding what I was talking about.

The first thing I learned was that my computer was dumber than a human being, but smarter than a programmer. I won't go into the other things I learned, because it won't do you any good. You have to learn them the same way I did, by reading good textbooks, going to classes and working with the computer. Suffice it to say that I finally began to know what I was talking about in all those computer-oriented technical manuals I was now writing. And with that knowledge came the realization that this microcomputer really was the tool I'd been saying it was. That silly gadget started keeping my checkbook up to date, telling me what my business loans were going to cost me, keeping track of my billable time, generating the bills and, finally, handling my word processing.

Words, Words, Words

Word processing. What an exquisite tool for a free-lance writer! It's the closest thing yet devised to direct translation of thought to hard copy. Move that paragraph? Sure. No more cut-and-paste. A few keystrokes and it's done.

But it didn't come easy. First, I purchased a black box printer. A slow, all caps machine, it worked well for printing bills, but was hardly what I needed for quality word processing. Then I got a Trendcom to check out word-processing programs in caps and lowercase. No point in spending a lot of money on a printer until I had usable software.

I finally found Word Weaver, an inexpensive program by Bob Huelsdonk (Huelsdonk Products, 15703 Midvale North, Seattle, WA 98133). It worked so well that I didn't even squirm when I plunked down $3800 for a Diablo 1640 terminal, chock full of options.

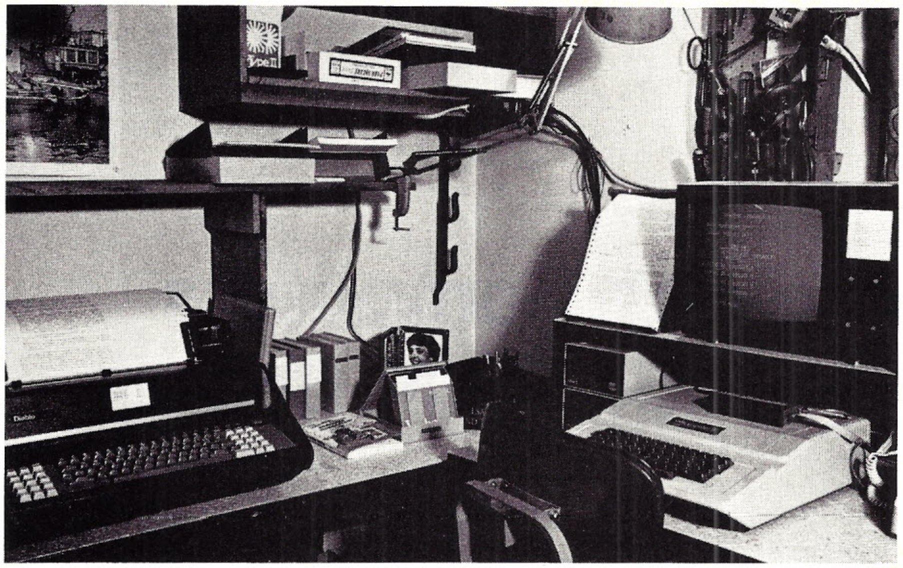
The System

My system included a 48K Apple with two disk drives, serial interface, Word Weaver and a Diablo. It spits out words three times faster than the Itel did in its prime and has yet to make a mistake. Not only does it let me keep up with my new-found workload and ask for more, it even gives me time to write articles and software.

For those of you who may be looking for a word processor, here's a rundown of my system:

1. A 48K Apple II, with Applesoft in ROM, the Paymar lowercase adapter, an Apple high-speed serial interface and a D.C. Hayes Micro modem. 32K is probably sufficient, but 48K is more convenient. The Micro modem has nothing to do with word processing, but some of my customers are interested in time-sharing on my computer.

2. A pair of Apple disk drives. A single drive is OK, but the second drive facilitates generation of backup text file copies. I keep



Word-processing station for the Goodfellows, Commercial Publications. I've been accused of using the tools mounted on the wall over the video monitor for fine-tuning the system. Not so, not so!

the program in drive 1 and write text to drive 2. This allows easy filing, with one diskette per manual. I have had as many as 50 pages on a single disk, using both sides, and have never run out of space.

3. A Leedex Video 100 12-inch B/W video monitor. I started out with a G.E. 12-inch portable TV, but the fuzzy picture was hard on my eyes. The extra 200 lines of resolution makes all the difference in the world when you're trying to read capital and lowercase letters on the screen.

4. A Diablo Model 1640 terminal, with word-processing enhancement. I bought a terminal instead of a printer because the resident keyboard lets me bypass the computer to directly type the figure captions, specially formatted page numbers, words to be pasted into illustrations and so forth. The word-processing enhancement option supports automatic underscore, shadow print, bold print, auto center and proportional spacing. My software takes care of auto center, so that feature is redundant.

I will make software changes to support other options, because I can't stand the thought of those features being there but in hiding. So far I haven't needed them, because my clients have been happy with the copy as produced.

5. A Sears cassette tape recorder. Once I used this to load programs.

6. Miscellaneous software, including Word Weaver by Huelsdonk, Client File Billing by me, Checkbook and File Cabinet by Apple and a host of smaller programs by various magazine contributors.

The computer has turned things around at our house. Its mere presence gives us a prestige I'm vain enough to enjoy, as well as brings in new business. It's turned my daughter (now 14) into a hotshot programmer, for better or worse. It gets the whole family involved in games. And it's increased my productivity by about 70 percent, thus paying for itself many times over in less than two years. It has even increased my capabilities, by allowing me to trade my tired old Itel for a tired old press. ■

Digital Research Computers 16K SS-50 Static RAM Board

Boost your memory without depleting your pocketbook.

Dennis Doonan
c/o Graphics I
345 Main St.
Racine, WI 53403

Like many hobbyists, I was faced with insufficient memory in my system and not enough money for additional boards. Also, my motherboard was running out of main slots. With three 8K RAM boards, the CPU, the Percom disk controller and I/O boards, the SWTP power supply was close to its limit.

What I needed was a low-power 16K board. But while SWTP, Gimix and Smoke Signal Broadcasting provided excellent boards that would fit my need, they did not fit my budget.

Fortunately, Digital Research Computers (PO Box 401565, Garland, TX 75040) came to the rescue with its 16K static RAM board for the SS-50 bus. Though a complete kit costs $229, the bare board is available for $30, the support ICs and capacitors for $19.95, and a complete set of sockets for $12. DRC sells eight low-power, 300 ns 2114 RAMs for $44 (4K worth).

So for $105.95 (or less, if you furnish the sockets and chips) you can assemble a 4K board that is expandable to 16K in 1K steps. Granted, you will save money by buying the complete kit, but you can start with 4K and expand as need and finances permit. This is a tremendous advantage for home computer users.

The board is good quality—double-sided, solder masked, with component labels. The sockets are a mixture of prime gold TIs, quality tin and less expensive AMP types.

The mix of parts had me wondering for a while. But top-quality parts are used in the critical areas, and adequate parts are used in the support areas. The 2114 chips proved to be an excellent value, so much so that I will soon be ordering more.

Assembly

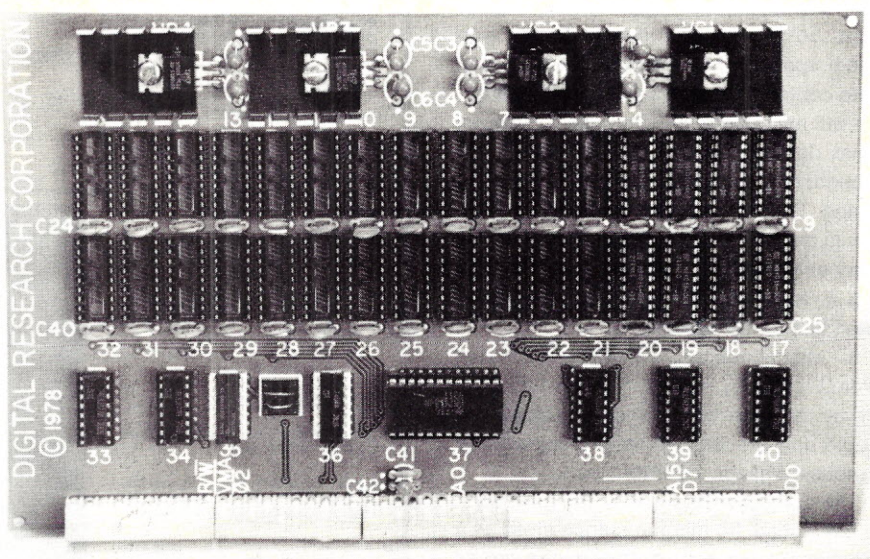
I assembled the kit and ran memory diagnostics in a single evening. Assembly was straightforward. The documentation is better than most manufacturers' and is topped only by the Heathkit manuals. For users past the beginner stage, documentation rates "very good."

Placement of the four voltage regulators along the top of the card to keep heat away from the memory chips is good designing. The only feature lacking is the write protect option of the SWTP 8K board. I must admit I have never used this option, so I don't miss it.

I've had the DRC board in daily use without a sign of trouble. It is addressed on 16K boundaries by jumpers. Since it cannot be split into 8K or 4K blocks, you must give some thought to its placement in your system's memory map. I placed mine in the 16K to 32K position. With only 4K worth of 2114s installed, BASIC crashed.

Most BASICs seek the end of memory automatically. Addresses without chips read as $00 rather than $FF of unassigned locations. Just supply the end-of-memory pointer with the actual ending address. In Percom Super BASIC this is location $0150.

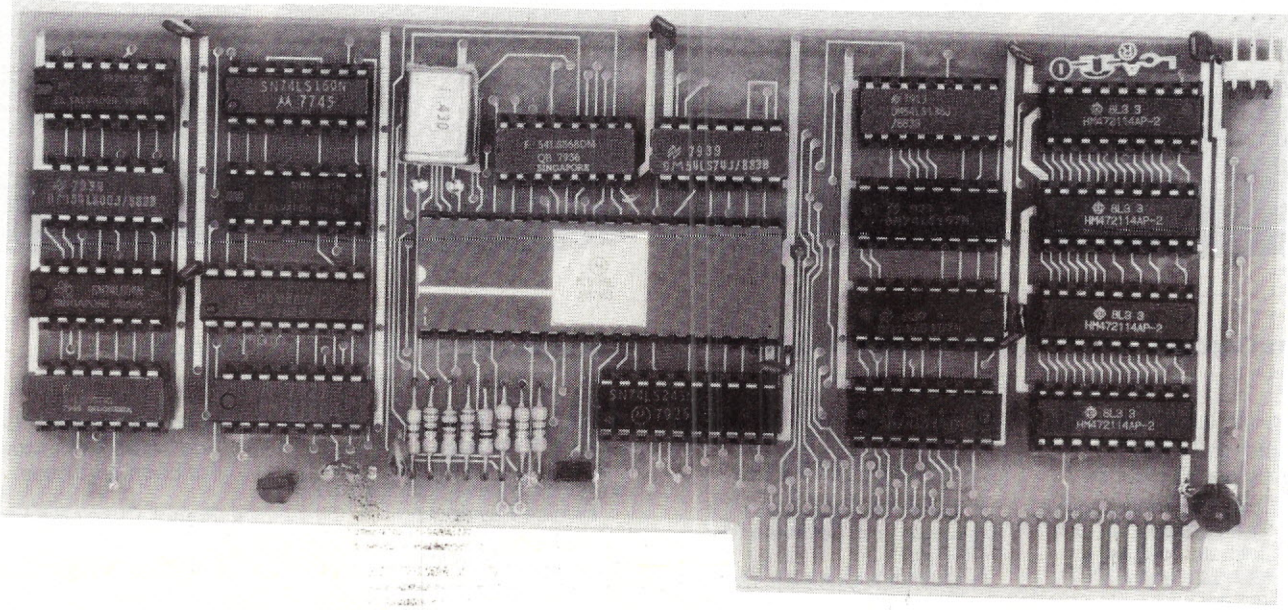
To sum up, the Digital Research Computers 16K board lets me add memory in affordable bytes, and I strongly recommend it.■



The DRC 16K board.

FOR APPLE II AND APPLE II PLUS COMPUTERS

DoubleVision™



80 x 24 Video Display with Upper and Lower Case
COLUMNS LINES

• is a hardware board that may be plugged into any slot in Apple II or Apple II Plus 32K or 48K Disks • full 128 ASCII character set, including control characters • fully programmable cursor • built in light pen capability • inverse video • full cursor control • works with 50/60Hz • has 2k of its own screen memory • has its own video output jack that must be connected to a monitor (or a high band width black & white TV thru a good RF modulator). Color TV's produce a poor display and are not recommended. • permits you to connect another monitor (or a T.V. set thru RFmod) to the Apple video output jack • displays 24 lines of 80 column text — programmable for different values • permits you to have graphics on Apple video output • video output and Apple video output may be connected to one monitor thru optional video switch • is active only when addressed for reading from or writing to • accepts lower case input from keyboard by use of escape key. (no modification required) or direct use of shift key (1-wire connection from shift key pad to DoubleVision required). • is compatible with the latest version of various word processing software packages. Presently these include Apple-pie 2.0— Programma International, Easywriter Professional system—Informational Unlimited, Text Editor/Formatter—Peripheral's Unltd. (when ordering from these companies, please ask for versions compatible with DoubleVision). All software available from Computer Stop when released. • Peripheral's Unltd. B.I.T.S. and P.I.T.S. and Southeastern Software's "DATA CAPTURE" with Micromodem and communication card. These packages give ability to upload, transfer and download files from remote computers, and all at 80 columns! • Programma Int. latest assembler LISA V:20 will support full 80 column display • is transparent for use with Basic and Pascal • software on disk for easy modification and adaptation for different applications • completely commented source listing of software and hardware schematics available • PASCAL (optional) • becomes the console when installed in Pascal • Permits 80 column text processing with full upper/lower case while using Pascal's editor • must be plugged into slot 3 when operating with Pascal

Available now at your local computer store....... $295.00

Call Computer Stop for Store nearest you

Shipping, Insurance, Handling, extra

Calif. Residents add 6% Sales Tax

*Apple is a Registered TM of Apple Computers, Inc.

Dealer inquiries invited.
Contact:

COMPUTER STOP CORP.
2545 West 237th St.
Suite L
Torrance, CA 90505
539-7671

The Computer Stop
16919 Hawthorne Blvd.
Lawndale, CA 90260 ✓283
(213) 371-4010

MON. - SAT.
10-6

Relocating the Dynamic Debugging Tool

CP/M owners will save troubleshooting time with this program.

Ken Barbier
Borrego Engineering
PO Box 1253
Borrego Springs, CA 92004

If you do much customizing of your CP/M operating system, this Relocate DDT program can save you a lot of trouble-shooting time. It will allow you to run CP/M's Dynamic Debugging Tool (DDT) in memory along with the entire CP/M operating system.

As originally supplied with CP/M, DDT loads itself into memory, overlaying the CP/M Console Command Processor (CCP). Digital Research created it this way to save memory space.

But if you want to make patches in the operating system or your Custom Basic I/O System (CBIOS), you have to go through a number of error-prone operations to move the entire operating system image down into empty RAM, make your patches and write the results onto the disk. Then you have to reload the modified system and test your changes. And you can't use DDT to help with this testing.

This procedure is more complicated than it needs be. An alternative, made possible by the Relocate DDT program, allows you to use DDT to make changes in your system as it normally resides in memory. DDT can then be used to debug the modified version of CP/M. When you have verified that your customized version is working, you can write it out onto the disk as your new operating system.

This last step requires that you have a routine in your disk operating system (DOS) or on disk as a .COM file, which will cause the operating system resident in memory to be written onto the disk beginning on track 0. Since this operation is highly hardware specific, you will have to supply it yourself if it is not included with your DOS. This program will be different for each version of CP/M and each computer.

Even if you do not have access to such a Write System to Disk program, Relocate DDT can still save you a lot of debugging time by allowing you to enter and test system changes through the DDT facilities. When your changes are fully checked out, then you can use MOVCPM and SYSGEN to save the updated system.

How the Relocator Works

When called from the console, DDT is loaded into RAM beginning at location 100, as are all transient programs (all addresses are shown in hexadecimal). Before beginning execution, however, DDT relocates itself in memory, taking up the 5K bytes just below the BASIC disk operating system (BDOS). In the 16K version of CP/M, DDT will move up to addresses 1800 through 30FF. Since the CCP resides at 2900 through 30FF, it will be wiped out by DDT as a result of this move.

In a 16K system, this overlaying is necessary for debugging user programs, since there isn't much user workspace available to begin with. But if you want to debug system changes, and not user programs, you'll have enough memory space and you won't want DDT to overlay the CCP.

When first loaded at location 100, DDT looks at the two-byte address portion of a jump instruction stored at location 0005 to determine how far up in memory to move.

```
2900 =          CBASE   EQU     2900H           ; BASE OF 16K CP/M CCP
4000 =          BIAS    EQU     4000H           ; OFFSET FOR 32K CP/M
68F0 =          NBASE   EQU     CBASE+BIAS-10H  ; NEW BASE ADDRESS

0100                    ORG     0100H

0100 2A0600     DDTX    LHLD    6               ; MOVE OLD JUMP TO NEW BASE
0103 22F168             SHLD    NBASE+1         ; (JUMP IS TO BDOS ENTRY)
0106 21F068             LXI     H,NBASE         ; SET NEW BASE ADDRESS
0109 220600             SHLD    6               ; INTO PAGE 0 JUMP
010C 36C3               MVI     M,0C3H          ; JMP OPCODE TO NEW BASE
010E 210344             LXI     H,4403H         ; CREATE NEW COMMAND
0111 220769             SHLD    CBASE+BIAS+7    ; IN CONSOLE INPUT BUFFER
0114 214454             LXI     H,5444H         ; SAYING "DDT"
0117 220969             SHLD    CBASE+BIAS+9
011A AF                 XRA     A               ; TERMINATED WITH 0
011B 320B69             STA     CBASE+BIAS+0BH
011E 3E08               MVI     A,8             ; RESET COMMAND POINTER
0120 328869             STA     CBASE+BIAS+88H
0123 3A0400             LDA     4               ; SELECT CURRENT DRIVE
0126 4F                 MOV     C,A
0127 C30069             JMP     CBASE+BIAS      ; AND LOAD DDT

012A                    END

0000 2A 06 00 22 F1 68 21 F0 68 22 06 00 36 C3 21 03
0010 44 22 07 69 21 44 54 22 09 69 AF 32 0B 69 3E 08
0020 32 88 69 3A 04 00 4F C3 00 69 00 00 00 00 00 00
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Program listing. DDTX program in assembly language.

This address field is part of an instruction that is a jump to the BDOS entry point and was stored here when CP/M was moved into memory by the bootstrap loader. Knowing where BDOS is, DDT can compute where its own start address should be relocated.

If you change that location 5 address field, you can force DDT to load itself anywhere in memory. But things are not all that simple, since DDT uses this jump instruction to access CP/M's I/O facilities.

You can move this jump instruction to another place in memory and substitute a "jump to *that* location" instruction in location 5. Now DDT will relocate itself just below the address where you put the moved instruction. Then when DDT calls location 5, it will encounter a jump to the moved instruction, which will, in turn, jump to BDOS at the proper entry.

While this sounds complicated, it only takes 42 bytes of code to set up these two jumps and to make other changes necessary to get DDT relocated and running. The program listing is called DDTX to differentiate it from DDT itself. From the command mode of CP/M, you call for DDTX instead of DDT. DDTX sets up the two jump instructions (see the first five lines of code) and then sets up the CCP input buffer to make it think you really asked for DDT.

Without going into needless detail, this requires loading a command length value (3), followed by the ASCII for DDT, followed by 0, all starting at a location seven bytes above the beginning of the CCP. Next you set a pointer to the beginning of this command.

All that's left then is to load the current disk drive number into register C and jump to the start of the command processor. CCP will look into its input buffer and find DDT; it will load and execute DDT, and you will have a relocated version of DDT.

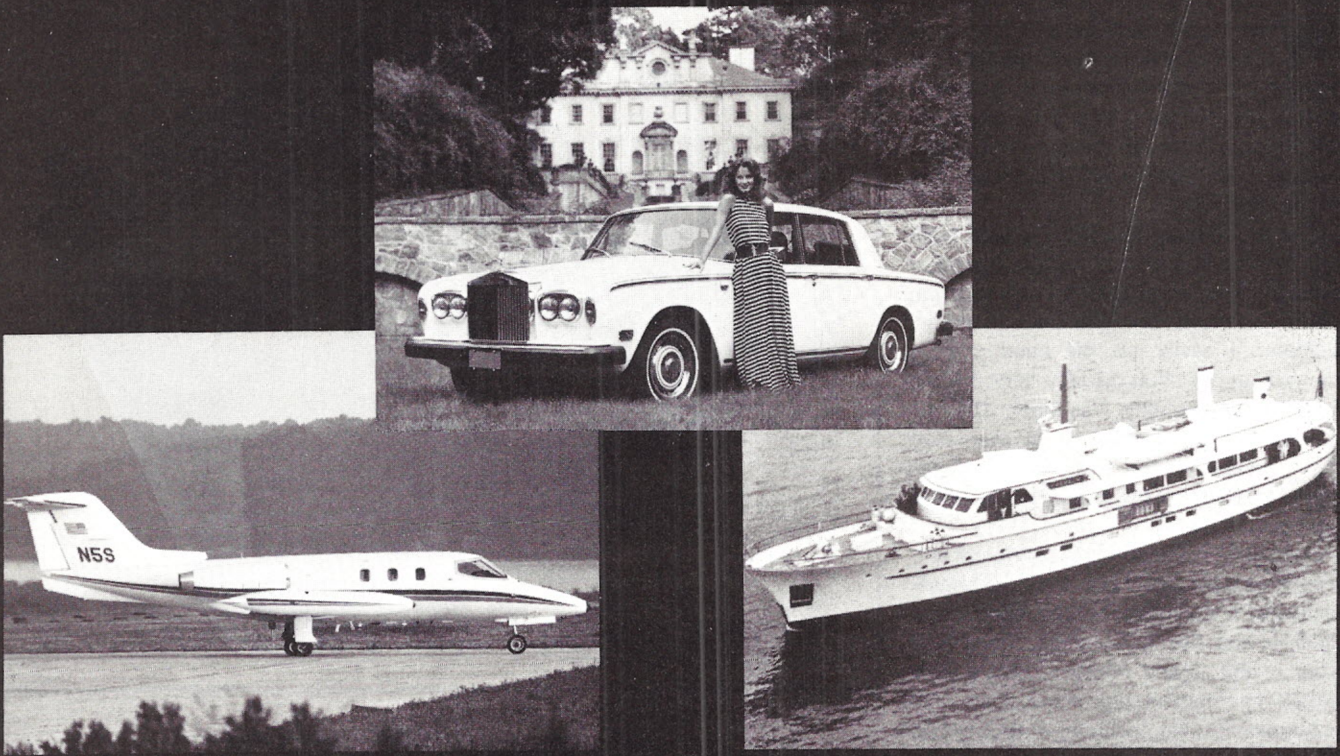
Understanding the Listing

This program is written in a general form to permit its use with any size version of CP/M. If you have relocated CP/M, you will know what BIAS has been added to the BASE address of CP/M to get it to the top of memory. In the listing, BIAS is set to 4000, for a 32K version. The value of BIAS in the pseudo-operation "BIAS EQU 4000H" is the only change you will have to make to use this routine with any version of CP/M. If your BIAS is different, change the value in this line and reassemble DDTX. The other addresses required will be computed by the assembler.

These other addresses include a new base address, NBASE, where you will place your moved instruction. The assembler similarly computes the addresses for the command line and its pointer and the entry to the command processor. ■

Hundreds of Rolls Royces, Yachts, Airplanes & Premium Properties

THE ROBB REPORT



THE ROBB REPORT, published monthly, is the market place for the buyer, seller and trader who appreciates the finer things in life. Listed for sale are hundreds of new and previously owned antique and classic motor cars, yachts, airplanes, premium properties horses, art, firearms and antique treasures. Complete descriptions and photographs are included, as well as the owner's name, address and telephone number.

A 12 month subscription to THE ROBB REPORT is $45.00*. Send your check to the address below. For even faster 24 hour service, call toll-free: 800-228-2606. (In Nebraska call 800-642-8777). For renewals or information call 404-256-9470.

The Robb Report

THE MAGAZINE FOR CONNOISSEURS

THE ROBB REPORT/P.O. Box 720317/Atlanta, Georgia 30328

*Add $100.00 for overseas subscription. Prices subject to change without notice.

A PLL UART Clock

The author enhances his 6800 system with a low-cost do-it-yourself clock synthesizer.

John M. Franke
1006 Westmoreland Ave.
Norfolk, VA 23508

When I finally built my 6800 microprocessor system, I found that I did not want to spend more money for a 1.8432 MHz crystal and MC14411 chip to generate the 2400 Hz clock for the Teletype PIA. I also needed a 16X clock or 4800 Hz clock to add to the 6850 ACIA.

The engineering note with the 6830 ROM recommended a simple resistor-capacitor network using the MC14536. I tried this with partial success. Using a ten-turn potentiometer, I easily set the baud rate by monitoring pin 13 on the MC14536. But the temperature stability was poor. The frequency had to be touched up every half hour; otherwise, the terminal would not print replies from the computer, or would print partial replies.

I decided to try a completely different and less expensive approach. I built a phase-locked loop to synthesize both 4800 Hz and 2400 Hz from the 60 Hz ac line. The synthesizer in Fig. 1 and Photo 1 uses a 4046 CMOS micropower phase-lock loop. The voltage-controlled oscillator output is buffered by an inverter, then divided by eight, and then by ten with a 4040 and 4518 to obtain a nominal 60 Hz.

The divided output is compared to the 60 Hz line frequency. The comparator output is fed through the low pass filter formed by the 100k and 27k resistors and 47 uF capacitor to the voltage-controlled oscillator. The loop locks in less than one second from power on and remains locked. The output from the locked oscillator is 4800 Hz; 2400 Hz is obtained from the first divider chip. Both outputs are buffered with 4049 inverters, which can directly drive two TTL loads each.

Since line frequency is held to better than one percent, the output stability is more than adequate. The 60 Hz could be supplied from a digital clock crystal time base if you desire total freedom from the ac line. The cost was under \$6 and the current drain is less than one mA. ■

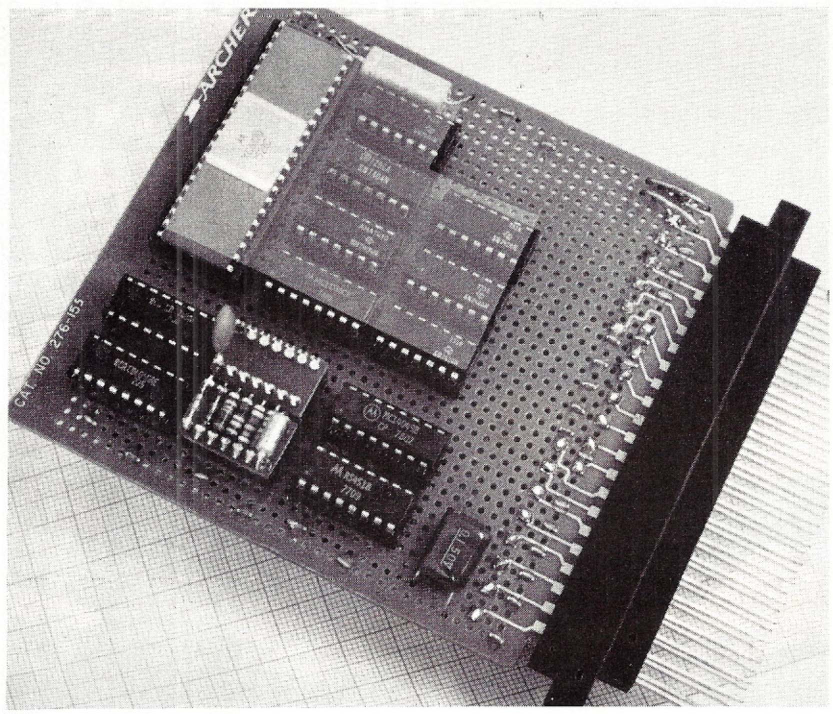
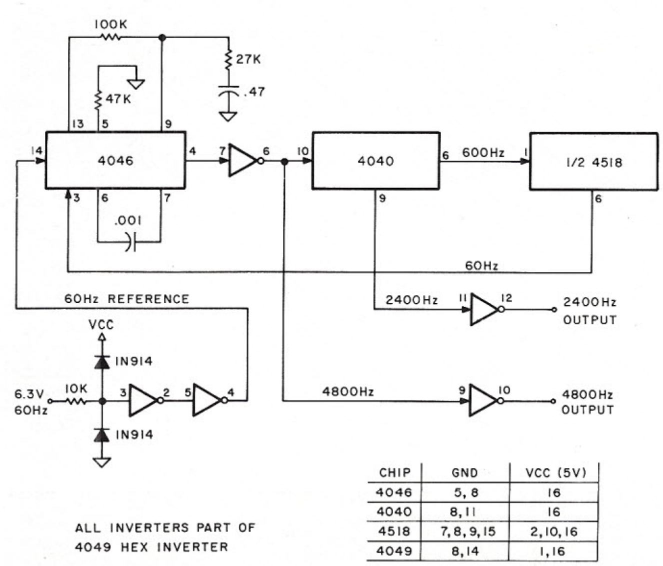


Photo 1. Phase-lock loop UART clock synthesizer.

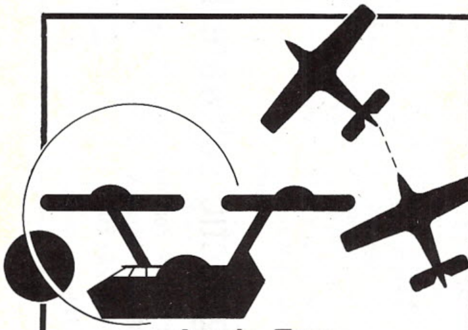


CHIP	GND	VCC (5V)
4046	5, 8	16
4040	8, 11	16
4518	7, 8, 9, 15	2, 10, 16
4049	8, 14	1, 16

Fig. 1. Synthesizer circuit.

Natural Organic Apple Software

Educational, intriguing and challenging...naturally!



Apple Fun

We've taken five of our most popular programs and combined them into one tremendous package full of fun and excitement. This disk-based package now offers you these great games:

Mimic—How good is your memory? Here's a chance to find out! Your Apple will display a sequence of figures on a 3×3 grid. You must respond with the exact same sequence, within the time limit.

There are five different, increasingly difficult versions of the game, including one that will keep going indefinitely. Mimic is exciting, fast paced and challenging—fun for all!

Air Flight Simulation—Your mission is to take off and land your aircraft without crashing. You're flying blind: on instruments only.

You start with a full tank of fuel, which gives you a maximum range of approximately 50 miles. The computer will constantly display updates of your air speed, compass heading and altitude. Your most important instrument is the Angle of Ascent/Bank Indicator. It will tell if the plane is climbing or descending and whether banking into a right of left turn.

After you've acquired a few hours flying time, you can try flying a course against a map or doing aerobatic maneuvers. Get a little more flight time under your belt and the sky's the limit!

Colormaster—Test your powers of deduction as you try to guess the secret color code in this Mastermind-type game. There are two levels of difficulty, and three options of play to vary your games. Not only can you guess the computer's color code, but it will guess yours! It will also serve as referee in a game between two human opponents. Can you make and break the color code...?

Star Ship Attack—Your mission is to protect our orbiting food station satellites from destruction by an enemy star ship. You must capture, destroy or drive off the attacking ship. If you fail, our planet is doomed.

Trilogy—This exciting contest of logic has its origins in the simple game of tic-tac-toe. The object of the game is to place three of your colors in a row into the delta-like, multi-level display. The rows may be horizontal, vertical, diagonal and wrapped around, through the "third dimension". Your Apple (or human opponent) will be trying to do the same, and there are many paths to victory. You can even have your Apple play against itself!

Minimum system requirements are an Apple II or Apple II Plus computer with 32K of memory and one minidisk drive. Mimic requires Applesoft in ROM, all others run in RAM or ROM Applesoft.

Order No. 0161AD $19.95

Paddle Fun

This new Apple disk package requires a steady eye and a quick hand at the game paddles! We've included four different games to challenge and amuse you. They include:

Invaders—You must destroy an invading fleet of 55 flying saucers while dodging the carpet of bombs they drop. Keep a wary eye for the mother ship directing the incursion. Your bomb shelters will help you—for a while. Our version of a well known arcade game! Requires Applesoft in ROM.

Howitzer—This is a one or two person game in which you must fire upon another howitzer position. This program is written in HIGH-RESOLUTION graphics using different terrain and wind conditions each round to make this a demanding game. The difficulty level can be altered to suit the ability of the players. Requires Applesoft in ROM.

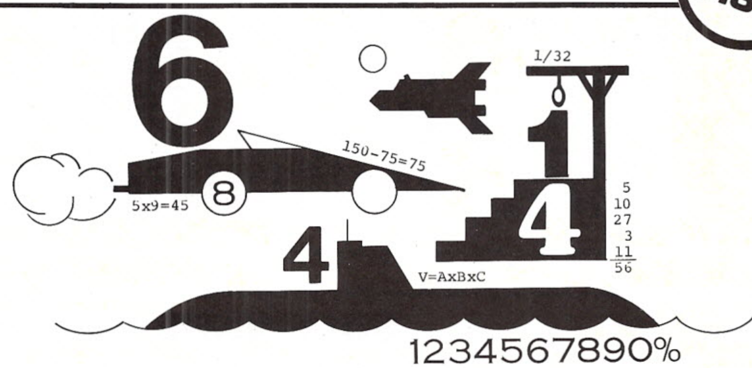
Space Wars—This program has three parts: (1) Two flying saucers meet in laser combat—for two players, (2) two saucers compete to see which can shoot out the most stars—for two players, and (3) one saucer shoots the stars in order to get a higher rank—for one player only. Requires Applesoft.

Golf—Whether you win or lose, you're bound to have fun on our 18 hole Apple golf course. Choose your club and your direction and hope to avoid the sandtraps. Losing too many strokes in the water hazards? You can always increase your handicap. Get off the tee and onto the green with Apple Golf. One of its nicest features is you'll never need to cancel a golf date due to rain. Requires Applesoft.

The minimum system requirement for this package is an Apple II or Apple II Plus computer with 32K of memory and one minidisk drive.

Order No. 0163AD $19.95

ON DISK



Math Fun

Change an Apple computer into a mathematics tutor and change boredom into enthusiasm with the Math Fun package. Using the technique of immediate positive reinforcement, students can improve their math skills while playing a game with:

Hanging—A little man is walking up the steps to the hangman's noose. But YOU can save him by answering the problems posed by the computer. The program uses decimal math problems. Each correct answer will move the man down the steps and cheat the hangman.

Spellbinder—You are a magician competing against a computerized wizard. In order to cast death clouds, fireballs and other magic spells on him, you must correctly answer questions about using fractions.

Whole Space—Pilot your space craft to attack the enemy planet. Each time you give a correct answer to the whole number problems posed by the computer, you move your ship. But for every wrong answer, the enemy gets a chance to fire at you.

Car Jump—Make your stunt car jump the ramps. Each correct answer will increase the number of buses your car must jump over. These problems involve calculating the areas of different geometric figures.

Robot Duel—Fire your laser cannon at the computer's robot. If you give the correct answer to problems on calculating volumes, your robot can shoot at his opponent. If you give the wrong answer, your shield power will be depleted and the computer's robot can shoot at yours.

Sub Attack—Practice using percentages as you maneuver your sub into the harbor. A correct answer lets you move your sub and fire at the enemy fleet.

All of these programs run in Applesoft BASIC, except Whole Space, which requires Integer BASIC.

Order No. 0160AD $19.95

TO ORDER: Look for these programs at the dealer nearest you. If your store doesn't stock Instant Software send your order with payment to: Instant Software, Order Dept., Peterborough, N.H. 03458 (add $1.00 for handling) or call toll-free 1-800-258-5473 (VISA, MC and AMEX accepted).

Instant Software™

Prices subject to change without notice.

PETERBOROUGH, N.H. 03458
603-924-7296

Universal Multiplexed Display

Here's one that's easy to operate and understand.

George Young
Sierra High School
Tollhouse, CA 93667

Computer buffs and hams are always displaying characters on seven-segment readouts. If the number of seven-segment displays used is four or five, it will usually be less expensive and simpler to use a nonmultiplexed circuit. At about six digits or more, or if multiple-digit displays are used, the circuits must be multiplexed.

Multiplexed displays are usually so complex that most of us will fight the extra lines running to the display unit rather than attempt to fight with the multiplexing circuitry to make it operational.

Fig. 1 represents a multiplexed display that is easy to operate and understand.

The clock was originally presented to *Kilobaud* readers in Kilobaud Klassroom #1, May 1977, and is formed from half of a 7404. The output is approximately 200 kHz with the .01 uF capacitor. For troubleshooting, this capacitor may be paralleled temporarily with a much larger capacitor. Try 10–100 uF to slow the clock down.

Operation

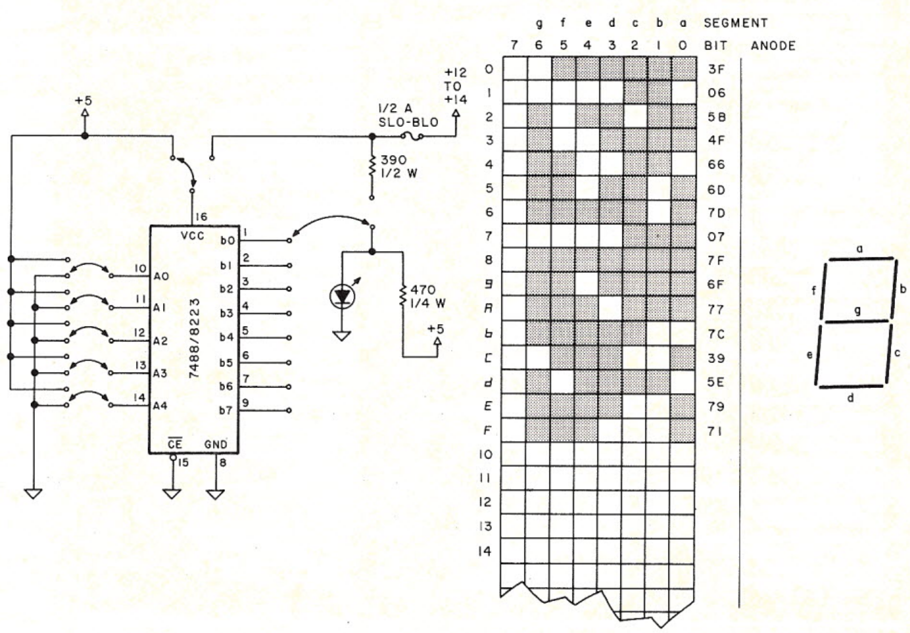
The clock output drives a counter. Fig. 1 shows a BCD counter, but a binary counter may be used. The counter can be from the 7490 family, the 74160 family, the 74176 family or the 74190 family.

The four outputs of the counter drive a 1:10 decoder, such as the 7442, to provide ten different active-low outputs. The decoder can be a 1:8 decoder, such as the 8250 or the 74155 connected as a 1:8 decoder, if fewer decoder outputs are needed. The decoder can be the 74154 used with a binary counter if more decoder outputs are needed.

A multiplexed calculator-type display is shown in Fig. 1. This can just as easily be discrete seven-segment readouts as well. Most multiplexed calculator readouts will require active highs on one side of the readouts, say for the digits, while the other side of the readouts will require the opposite type of drive, or active lows, for the segment drive.

One particular seven-segment multiplexed display, the Fairchild FNA 45, requires active-high drive for both the digits and the segments. For this reason inverters are shown between the decoder outputs and the digit drive inputs. Most seven-segment displays will not require these inverters.

A four-line to seven-segment decoder is used to drive the segments. For a common-cathode-type digital-only display, this would be the 7447 or an equivalent. For a common-anode-type digital display, this would be the 7448 or its equivalent. For use as a microcomputer display, this could be the Signetics 8T51 or the National DM8880 for common-anode-type displays. If your data manuals are a little more recent than



+5 volts and ground are applied to the chip for normal operation and for verifying the programmed bits. The chip is open collector, so pull-up resistors (470 ohm to 4.7k ohm is satisfactory) are required. The burn circuit for use on a solderless breadboard and matrix for using the 8223 as a hexadecimal decoder are shown. To burn a bit high (they are all low to start), address the row the bit occupies by jumping the address lines to ground or to +5 volts. Address the bit in the row by taking the bit line to +12 volts through a 390 ohm resistor. Raise Vcc (pin 16) to +12 for 10 ms; return Vcc to +5 volts. Connect bit line to LED test circuit. If the bit is now high, the LED will illuminate. If LED does not illuminate, bit is still low and burn must be repeated. Once high, a bit is permanently high.

Fig. 1. Simplified burn procedures for 7488/8223.

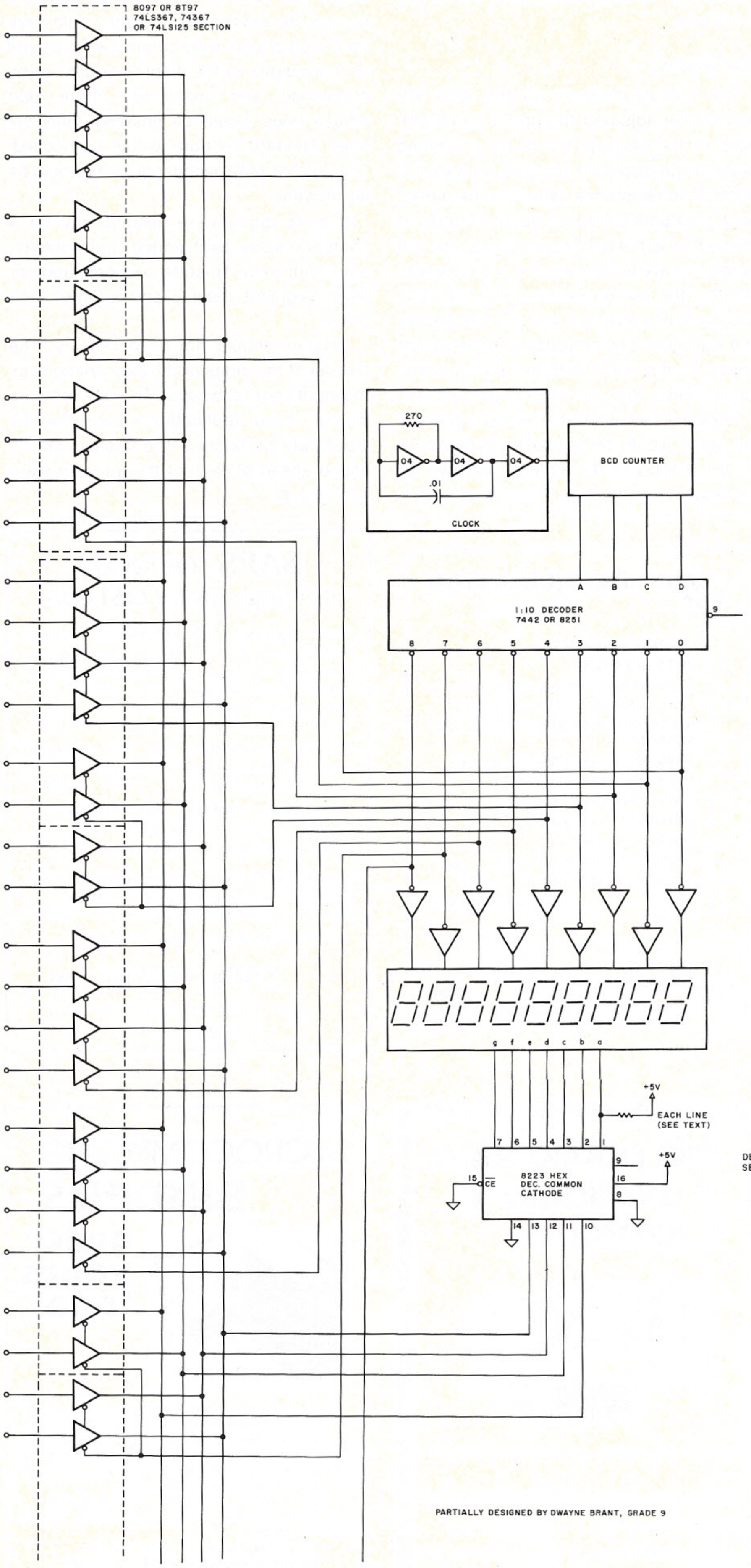


Fig. 2. "Universal" multiplexed display. (Partially designed by Dwayne Brant, Grade 9.)

mine, there are probably seven-segment decoders for the common-cathode-type displays available as well.

A 7488/8223 can also be programmed for use as a computer seven-segment decoder. Fig. 1 shows how to do this. Fig. 2 shows the 8223 used for this purpose, while Fig. 3 gives the burn pattern in matrix form to make the computer display decoder from the 8223.

The active-low outputs from the 7442 also are used as enables for Tri-state sections. The Tri-state sections may be any of the Tri-state buffers (the 8T97, 74367, 8097, 74LS367, 74125, 74LS125). The outputs of the Tri-state sections go to a four-line data bus, which feeds the inputs of the seven-segment decoder. Here it is shown going to four of the five address inputs of the 8223.

If pin 14, the A4 address input of the 8223, is low, the first, or upper, half of the matrix of Fig. 2 will be accessed. If pin 14 is tied to + 5 volts, the lower half of the matrix will be accessed. Both types of seven-segment decoders are shown in the matrix of Fig. 2, so that the circuit can be used with either common-anode-type displays or common-cathode-type displays. To select the decoder type, pin 14 is either low or high.

Now, whatever is on the inputs to the top four lines will be placed on the four-line data bus when the 0 output of the 7442 goes low. Digit 0 is simultaneously enabled. The

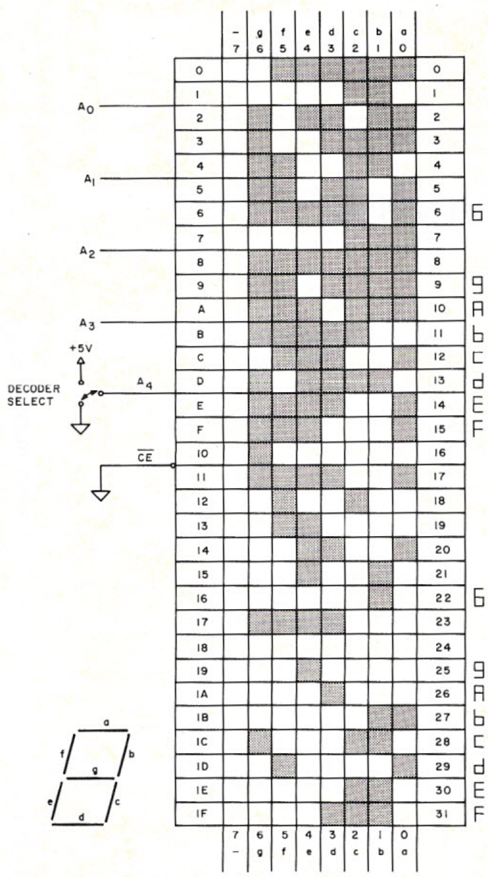


Fig. 3. "Universal" seven-segment decoder.

8223 decodes the data on its four input pins and activates the corresponding segments of the least significant digit on the display. Since no other digit enables are active, no other digits are illuminated.

A fraction of a second later, the 1 output of the 7442 goes low. The second digit is enabled, and at the same time the enable line of the second Tri-state group goes low, placing the inputs to this section on the four-line data bus to be decoded by the 8223 and turning on the appropriate segments.

When the nine digits shown in Fig. 1 have been scanned and illuminated, the 7442 will have one more output, which does nothing, and then the entire process will repeat. The display is being scanned so rapidly that the eye interprets the display to be "on" continuously. If the clock timing capacitor is paralleled with a large capacitance value, the clock will slow down enough to allow the builder to see exactly how the circuit functions.

The Display

The pull-up resistors on the 8223 will need to be adjusted to control the brightness of the display. Try 1k for a starting point. If the display is too dim, then try 470 ohms. The resistance value can be halved each time until you reach about 47 ohms. If the display is not bright enough yet, you will probably have to increase the +5 volts supplied to the pull-up resistors on the 8223. Use extreme caution if you run the display at a very slow rate for troubleshooting or for seeing how it works with low pull-up resistance values. You can destroy the display.

For use as a computer display, only six of the digits would be normally needed: four for the display of the address and two for the display of the data. The leftmost digit would not have its digit enable line connected; the next two to the left would have their digit enable lines connected; then one or more digits in the display would not be connected; and finally four more digit enable lines would be enabled. Only six quad Tri-state sections would be needed. The unused 7442 decoder outputs would be left floating.

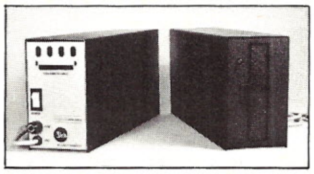
If more digits were needed, say for a counter display, each added digit would require four more Tri-state sections, another seven-segment display and another digit driver.

Latches should not be necessary. The display of the digits or the data changes as the input lines to the quad Tri-state sections change. If your application requires that the input data be latched, the latches can be added ahead of each of the quad Tri-state sections. ■

SIRIUS 80+

High Performance Low Cost Floppy Add-Ons!

The SIRIUS SYSTEMS 80+ Series of Floppy Disk add-ons are designed to provide unmatched versatility and performance for your TRS-80*. Consisting of four different add-ons, there is a 80+ Series Floppy Disk Drive to meet your needs.



COMMON CHARACTERISTICS

- *5ms track-to-track access time*
- *Auto-Eject*
- *180 day WARRANTY*
- *Exceptional speed stability - 11/2%*
- *Single/Double Density operation*
- *Mix any or all 80+ Series on the SS Standard cable*

SPECIFIC CHARACTERISTICS

The **SIRIUS 80+1** -a single sided, 40 track Drive. Offering 5 more tracks than the Radio Shack model, it cost $120 less. Formatted data storage is 102K/204K Bytes Single/Double Density.

SIRIUS 80+1 .... **$379.95**

The **SIRIUS 80+2** is a dual sided, 80 track (40 per side) Disk Drive. It appears to the TRS-80* as TWO 40 track drives yet **COST LESS THAN HALF THE PRICE!** Even greater savings result since data is recorded on both sides of the media instead of only a single side. This unit may require the SS Standard cable. Formatted data storage is 204K/408K Bytes Single/Double Density.

SIRIUS 80+2 .... **$449.95**

The **SIRIUS 80+3** - a single sided, 80 track Drive. Offering 2⅓ times the storage of a standard Radio Shack Disk Drive, the 80+3 greatly reduces the need for diskettes correspondingly. Additionally, because of the increased storage and faster track-to-track access time, the 80+3 allows tremendously increased throughput for disk based programs! The 80+3 includes SIRIUS's TRAKS-PATCH on diskette (for use with 96 tpi drives). Formatted data storage is 204K/408K Bytes Single/Double Density.

SIRIUS 80+3 .... **$499.95**

The **SIRIUS 80+4** -a dual sided, 160 track (80 per side) 5¼" monster! The ultimate in state-of-the-art 5¼" Floppy Disk Technology, the 80+4 is seen by the TRS-80* as two single sided disk drives. Thus, in terms of capacity, one 80+4 is equivalent to 4⅔ standard Radio Shack drives — at a savings of over 73% (not to mention diskettes!!!). (With a double density converter the available memory is huge!) The 80+4 (a 96 tpi drive) includes TRAKS-PATCH on diskette and may require the SS Standard cable. Formatted storage is 408K/816K Bytes Single/Double Density.

SIRIUS 80+4 .... **$649.95**

All 80+ Series Floppy Disk add-ons operate at 5ms track-to-track but are Expansion Interface limited to 12ms for the TRS-80*

**TRS-80© of Tandy Corp.*

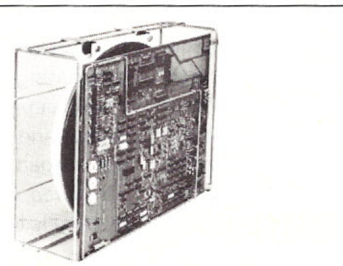
ACCESSORIES

SS Standard 2 Drive Cable **$29.95**
NEWDOS/80-Sophisticated Operating System for the TRS-80* from Apparat **$149.95**

Save up to 10% *with these SIRIUS Packages!*

NEWDOS/80, SIRIUS 80+3, and Two Drive Cable	**$624.95**
NEWDOS/80, SIRIUS 80+4, and Two Drive Cable	**$749.95**
NEWDOS/80, Two (2) SIRIUS 80+3's, Two Drive Cable	**$1080.95**
NEWDOS/80, Two (2) SIRIUS 80+4's, Two Drive Cable	**$1349.95**

PRIAM Hard Disks Now Available from SIRIUS SYSTEMS!



PRIAM's high-performance, low-cost Winchester disc drives speed up throughput and expand data storage from 20 megabytes to 154 megabytes. And a single controller can be used to operate 14-inch-disc drives with capacities of 33, 66, or 154 megabytes or floppy-disc-size drives holding 20 and 34 megabytes. So it's easy to move up in capacity, or reduce package size, without changing important system elements or performance.

- *Fast, Linear Voice Coil Positioning*
- *10 ms track-to-track positioning*
- *Fully servoed head positioning*
- *Dedicated servo tracks*
- *DC Power required only!*
- *Simple, parallel Interface*
- *Optional SMD Interface*
- *50 ms Average Positioning time*
- *90 ms Maximum Positioning Time*
- *6.4 ms Average Latency*

THE PRIAM LINEUP

Model/Disc Size	Capacity	Size	Weight	Price
DISKOS 3350 (14")	33Mbytes	7" × 17" × 20"	33 lbs.	$2995
DISKOS 6650 (14")	66 Mbytes	7" × 17" × 20"	33 lbs.	$3749
DISKOS 15450 (14")	154 Mbytes	7" × 17" × 20"	33 lbs.	$4695
DISKOS 2050 (8")	20 Mbytes	4.62" × 8.55" × 14.25"	20 lbs.	$2995
DISKOS 3450 (8")	34 Mbytes	4.62" × 8.55" × 14.25"	20 lbs.	$3745
DISKOS 1070	10.6 Mbytes	floppy-size	(low)	$2195

All PRIAM DISKOS Drives have a Transfer Rate of 1.03 Mbytes/Sec.
Optional SMD interface available for $150.

SIRIUS SYTEMS offer cases and enclosures for all PRIAM Hard Disk Drives. All 14" Winchester Drives will mount in our 14" Standard Case. The 8" Winchesters have two alternatives: a single drive case and a dual drive case. All SIRIUS SYSTEMS Winchester drive cases include Power Supply, internal cabling, switches, fan, extra AC outlet (not switched, but fused) and possess very adequate ventilation. Drive addressing is done on the rear of the Case and not on the drive iteself to provide ease of use during operation. All WINCHESTER DRIVE Cases are Warranted for a full year and come in our standard blue-black color scheme. Consult us for current availability and pricing.

Introducing the Versatile, Low-Cost OMEGA Series Controller

As new technological advances bring down the cost of fast, reliable mass data storage, the need for an inexpensive, versatile controller have become greater and greater. To meet this need, SIRIUS SYSTEMS' OMEGA Series Controller was designed.

The SIRIUS OMEGA Series Controller Module utilizes an **on-board microprocessor** to mediate data transfer to a wide variety of peripherals from an equally wide variety of host computer systems. Up to four Winchester Hard Disks (8" or 14"), four 5¼" Floppy Disks Drives and/or up to eight 8" Floppy Disk Drives may be **in use at one time.** Host systems interfacing is accomplished via a parallel or a serial interface. With the additon of a Personality module, the OMEGA Series Controller Module is directly **compatible with many popular computer systems** (among them the TRS-80*, Apple, Heath, and others). Provision is made for the addition of a streaming tape drive, also.

SPECIFIC HARDWARE FEATURES INCLUDE:

- *Control of up to twelve Floppy Disk Drives (eight 8" and/or four 5¼")*
  - *8" and/or 5¼" Disk Drive Utilization*
  - *Single (FM) or Double (MFM) density data storage*
  - *Hard or Soft sectored diskette usage*
  - *Utilization of "Quad" density (96 tpi) 8" or 5¼" Disk Drives*
- *Control of up to four WINCHESTER type PRIAM DISKOS Disk Drives*
  - *8" or 14" may intermix on the same cable*
  - *Accommodates 8" and/or 14" drives of 5.3Mbytes to 154Mbytes*
  - *Ultra-Fast data transfers*
- *Extremely flexible host-controller interfacing*

SPECIFIC SOFTWARE FEATURES INCLUDE:

- Dynamic format modifications via command words
- Extremely flexible format acceptance for unusual data storage formats
- Easily interfaces to standard operating systems (TRS-DOS*, CP/M®, etc)
- Operates in either get/put sector mode or data string mode
- Performance parameters may be changed by EPROM replacement or Dynaminic Reprogramming

CP/M® of Digital Research

Dedicated systems cards are also available on a limited basis for the STD-BUS and the S 100. These cards feature shared memory also (again, software selectable) in addition to the regular OMEGA Series Controller Module features. Consult SIRIUS SYTEMS for current price and availability for the entire line of OMEGA Series Memory Units and Controllers. **Dealer inquiries are invited.**

What TFORTH is - and what it has to offer YOU!

TFORTH is a unique growth programming language for the TRS-80® that combines the best features of an interpreter and a compiler all in one functional easy-to-use package. TFORTH cannot be simply compared with Fortran, BASIC or PASCAL. This high speed, high level modular code offers the speed found in many FORTRAN compliers yet retains the on-line conviences found in BASIC INTERPRETERS by flagging input errors as they occur line-by-line. Unlike PASCAL, TFORTH needs no "run-time" package for support. Serving as an operating system, compiler, assembler, interpreter, virtual memory manager, all in one: TFORTH makes easy, efficient- structured re-entrant programs a natural consequence.

The key to TFORTH's flexibility and ease of use lies in its use of a stack for parameters and a unique dictionary for WORDS. These WORDS are stated in terms of other WORDS already defined in the dictionary. It is this rich set of WORDS that provides DO LOOPS, IF-THEN-ELSE statements, BEGIN-END statements, virtual memory, any number base (to base 32) for input or output, a macro assembler, re-entrant code, multithread dictionary, line editor, excellent math package (16 bit integers, double precision floating point, SIN, COS, TAN, EXP and LOG) and it runs under either TRSDOS* or NEWDOS. Assembler inherently nests with high level in an easy fashion. Complicated drivers for new devices take only a few lines of TFORTH which saves both memory and disk space!

TFORTH is a procedural language specifing a process rather than a desired result. The ability to have the language grow in the direction the user desires is excellent for novel applications. New data types and new processes can become part of the language. Due to the modular constructions, a very compact code is produced which executes at exceptionally high speeds between machine code and machine code plus 20% typical overhead speeds. Memory requirements can be "less" than assembler coding or other high level languages.

TFORTH comes complete for the TRS-80® with as little as 16K of memory and a single Disk Drive using either TRS-DOS® or NEWDOS. It provided on diskettes and an optional Math and Utilities package is available.

Through TFORTH an excellent way to develop new languages, provide simple control of device (including video monitors, A/D and D/A converters and burglar alarms) and to implement tasks requiring monitoring and decision is offered. Many WORDS to handle peripherals are part of basic TFORTH and others may be added easily. Often, substantial hardware development can be eliminated by using TFORTH to do the major digital or reduction of data.

For many applications a minimal task may be written in high level (or mixture of assembler and high level) code: loaded, assembled and prior to execution may be written to the disk as a ready to execute machine code/EXE module with the DOS.

TFORTH (on diskette - specify for Standard or 96 tpi Disk Drives) **$129.95**
TFORTH with the addition of TRAKS-PATCH (a powerful combination!) **$136.95**

STATE-OF-THE-ART DISK DRIVES

QUME® DataTrak 8

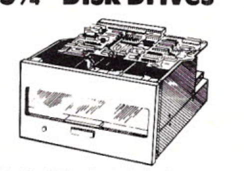
8" Disk Drive DOUBLE SIDED! DOUBLE DENSITY!

High performance Double Sided Disk 8" Disk Drive ■ Single or Double Density ■ Door Lock and Write Protect INCLUDED! ■ Negative DC Voltage not required ■ Low Power Operation

- *FAST! 3ms track-to-track access*
- *Low friction and minimum wear*
- *Superior Head Load Dynamics*

QUME DataTrak 8 .... **$574.95** **(2/$549 ea)**
QUME Technical Manual **$6.95**
Connector Set #3 (AC, DC, & Card Edge) **$10.95**
Connector Set #4 (AC and DC) **$2.95**

MPI 51/52 & 91/92 5¼" Disk Drives



- *Fast! 5ms track-to-track access*
- *Exclusive Pulley-Band Design*
- *Unique Door/Ejector Mechanism*
- *Reliable 11/2% Speed Stability*
- *Single/Double Density Operation*
- *Industry/ANSI Standard Interface*

MPI 51 (Single Head/40 tracks)
125K/250K Bytes Single/Double Density** **$259.95**

MPI52 (Dual Head/80 tracks (40/side))
250K/500K Bytes Single/Double Density** **$349.95**

MPI 91 (Single Head/80 tracks)
250K/500K Bytes Single/Double Density** **$399.95**

MPI 92 (Dual Head/160 tracks (80/side))
500K/1000K Bytes Single/Double Density** **$524.95**

MPI Technical Manual **$6.95**
**Unformatted data storage

↙67

SIRIUS SYSTEMS
7528 Oak Ridge Highway
Knoxville, Tennessee 37921

TO ORDER CALL (615) 693-6583

Phone Orders Accepted 9AM-7PM (EST) Mon-Fri

We accept **MC, VISA, AE, COD** (requires Certified Check, Cashier's Check or Cash) and **Checks** (personal checks require 14 days to clear). **SHIPPING AND HANDLING:** $7.00 per Floppy Disk Drive or 80+ Module ■ 5% for other items (any excess will be refunded) ■ **Foreign Orders** add 10% for Shipping & Handling. Payment in U.S. currency ■ Tennessee residents add 6% Sales Tax ■ **VOLUME DISCOUNTS AVAILABLE**

Electronic Systems Serial I/O Interface Kit For the Apple

Inexpensively converts an Apple II to a terminal.

Edward Burlbaw
945 Brook Circle
Las Cruces, NM 88001

I ordered an Electronic Systems serial I/O interface kit recently to inexpensively convert my Apple II to a terminal. I figure I couldn't go wrong for $42. I haven't regretted the choice.

I panicked when the kit first arrived—it had no step-by-step assembly instructions. But I found that instructions weren't necessary. The package included a schematic diagram, a circuit description and a photograph of the completed board. Also, the part values were silk-screened on the circuit card. I was able to "stuff" the board in less than one hour.

I didn't need fancy equipment for the baud rate adjustment—Electronic Systems included a baud rate adjustment program for just that purpose. The program showed the current baud rate digitally on the video screen, and also provided a relative analog display to guide me to the correct adjustment.

The kit contained parts necessary to build a 110 baud interface and instructions to change the circuit for rates up to 2400 baud. I needed 300 baud, so I had to change one capacitor. Switch-selectable baud rates would be possible with an extra switch or two, but that was too fancy for my needs.

On board, a DIP-socket-type bank of five switches selects parity, on or off, odd or even, number of stop bits and number of data bits, as required by your application. The terminal software (included) is written for peripheral slot #0, but it is easy to adapt the software for other slot usage with the information E.S. gives on the Apple's peripheral connector memory locations. The software lets you use the Apple as either a "dumb" or "intelligent" terminal, and you can use a Teletype as input and/or output. Software for output in correspondence code is also included.

The I/O board is advertised as RS-232 input and output. The enclosed literature contains a schematic for an RS-232/20 mA current loop interface (junk-box type of parts).

E.S. fails to mention one additional feature. I discovered it by necessity when I purchased the E.S. modem kit, intending to hook it to the I/O board. I failed to notice that the modem was TTY compatible and the I/O board was RS-232. I was able, though, to squeeze out of the board the required TTL input and output. The UART chip is TTL and the output is RS-232, so somewhere TTL is converted to RS-232. All I had to do was locate the input (and output) of the conversion circuit. Since E.S. also sells an RS-232 TTL conversion board, the schematic was in the E.S. catalog enclosed with my order. The TTL input and output are pins 20 and 25, respectively, of the UART chip.

Convenient places on the circuit board to solder the TTL leads needed to be found. I used a feed-through hole in the board for the output, while a 1k resistor lead made a convenient solder point for the TTL input. Now I was ready to directly couple to the modem.

While I was at it, I tapped the +5 V off the I/O board to power the modem, thereby eliminating the need for an additional power supply. I also jumpered the RS-232 output and input. The character being output echoed back to the video terminal. This will force half-duplex mode whether the modem is in half or full-duplex mode operation.

Building the serial I/O interface kit was easy. I was temporarily confused by the addition of a data terminal ready (DTR) connection with its associated components that was not mentioned in the included schematic. I assume that it was new to the board, and that E.S.'s literature had not been updated. DTR is not used for the simple modem application. I'm sure that if you had a data terminal with a DTR line, you would appreciate its inclusion.

If you are still uncertain about your kit-building abilities, you might consider buying the assembled and tested board for an additional $20. Electronic Systems offers to repair a nonworking board for $10, so even if you botch the job you can still come out ahead.

Whichever route you choose, you will have an inexpensive but versatile I/O board for your Apple, and will be one step closer to hookup with a CBBS Electronic Systems Serial Interface. ■

OHIO SCIENTIFIC

means BUSINESS . . .

with years more experience in hard disk technology, time sharing and networking, OHIO SCIENTIFIC is the state of art in computer technology.

SO DO WE!

1. ***Service.*** **Factory trained personnel with years of experience. Service contracts available.**
2. ***Full-Line Dealer.*** **Visit our showroom. We have nearly the entire OSI line in stock. (Both Business & Personal Computers).**
3. ***Software Support. If its for OSI, we've got it plus we've tested it.***
4. ***OEM/Softwarehouse Support. We don't just sell and forget. We help you make your sale.***

✓38

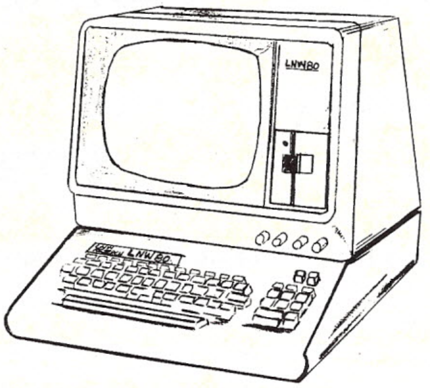
INTECHNOLOGY SERVICE ORG.
23 East 20th Street
NEW YORK, NY 10003
(212) 673-6310

APPLE II PLUS WITH 48K RAM	**$1190.**
TEXAS INSTRUMENT 99/4 COMPUTER WITH MONITOR	**$ 989.**
TI810 PRINTER	$1590.
CENTRONIC PRINTERS:	
730-1 PARALLEL PRINTER	$ 659.
737-1 PARALLEL INTERFACE	$ 879.
SAVE ON ALL OTHER MODELS	
SPINWRITERS FROM NEC	
5510 R/O + 5530 R/O	$2490.
5520 KSR SERIAL WITH KEYBOARD	$2795.
PAPER TIGER 440G	**$ 990.**
960	**$1149.**
BASE-2 PRINTERS 800 M.S.T.	**$ 649.**
ANADEX 9501	**$1390.**
COMPRINT 912 APPLE, TRS-80, PET	**$ 559.**
912 SERIAL	**$ 599.**
SYM-1 W/MANUALS	**$ 229.**
COMMODORE BUSINESS MACHINES:	
CBM 8032	$1595
PET 2001-16K	$ 895.
PET 2001-32K	$1090.
PET 2022 TRAC. FEED PRINTER	$ 749.
PET 2023 FRIC. FEED PRINTER	$ 679.
PET 2040 DUAL FLOPPY DISK DRIVE	$1090.
PET 8050	$1499.
ATARI 800 16K PLUS FREE 8K	**$ 799.**
INTERTEC SUPERBRAIN (32K)	**$2595.**
NORTH STAR COMPUTERS	
HRZ-2-32K-D-ASM	$2275.
HRZ-2-32K-Q-ASM	$2675.
DISPLAY TERMINALS:	
INTERTUBE II	$ 775.
HAZELTINE 1410	$ 775.
HAZELTINE 1420	$ 899.
TELEVIDEO 920C	$ 849.
SAVE ON COMPLETE HAZELTINE LINE	

IMMEDIATE DELIVERY FROM STOCK PRICES SUBJECT TO CHANGE
MULTI-BUSINESS COMPUTER SYSTEMS
28 MARLBOROUGH STREET
PORTLAND, CONN. 06480 ✓81
M/F 9-6 SAT 9-3 (203) 342-2747
TWX: 710-428-6345
MBCSYS

master charge VISA

THE FIRST TRS-80® COMPATIBLE COMPUTER WITH HIGH DENSITY COLOR GRAPHICS!



LNW RESEARCH introduces the LNW80, a high performance color computer, compatible with the TRS-80™ Model I. The fully integrated LNW80 is a sophisticated and versatile microcomputer with the following powerful features.

COMPATIBILITY
Hardware and software compatible to the Radio Shack TRS-80™ Model I computer, provides the widest software base of any microcomputer. Cassette interface; expansion bus.

DISPLAY
Quality upper and lower case display.

Two modes of color graphics, high resolution graphics, 384 × 192 in eight colors—higher density than the Apple II.* Low density color graphics of 128 × 192 are also available in eight colors.

High resolution—black and white graphics—of 384 × 192 mixed with text and TRS-80™ standard graphics.

Reverse video, composite video RF output.

PERFORMANCE
The LNW80 utilizes the fast Z-80A microprocessor which executes at a speed of 4 MHZ—over twice the speed of the TRS-80™ Model I.

LNW80

PC BOARD $89.95

Ask about our : Keyboard cabinet **Leedex VIDEO 100-80**

LNW RESEARCH

✓198

LNW RESEARCH 3183-E AIRWAY AVE COSTA MESA CA 92626 714-552-8946

*Apple II is a TM of Apple Computer, Inc. TRS-80 is a TM of Tandy Corp.

NEW EXTERNAL DATA SEPARATOR
ASSEMBLED AND FULLY TESTED
$14.95
SOME SOLDERING REQUIRED

SYSTEM EXPANSION

AT $69.95 [PC BOARD & USER MANUAL]

- SERIAL RS232C/20 mA I/O
- FLOPPY CONTROLLER
- 32K BYTES MEMORY
- PARALLEL PRINTER PORT
- DUAL CASSETTE PORT
- REAL-TIME CLOCK
- SCREEN PRINTER BUS
- ONBOARD POWER SUPPLY
- SOFTWARE COMPATIBLE
- SOLDER MASK, SILK SCREEN

ORDERING INFORMATION

Add $3 for postage and handling.
CA residents add 6% sales tax

VISA master charge

Video the Easy Way

The Gimix Ghost Video Board for the SS-50 bus.

Jerry Sorrels
6266 Banner Ct.
Riverside CA 92504

As a confirmed hardware person, I was reluctant to trust a video display that used software to do the scrolling and cursor positioning. Sure, I know there are lots of this type in operation, but not in my system! For the application I was working on, however, I had to keep it simple and the cost and size down.

I started looking at the different SWTP-compatible video boards. One that looked promising was the Gimix Ghost Video Board. I called Gimix for more information and discovered that they make an entire line of SWTP compatible boards, including their own 6800 system.

The price of the video board had just been reduced by $51 to $198 for an assembled and tested board, including shipping.

I was going to order the board immediately, but Gimix Vice-President Richard Don suggested that if I wasn't in a hurry he would send the documentation so I could make sure the board would be suitable. Two days later I had the information. Not bad for Chicago to California!

I received the video board's instructions, including a simple 84 byte driver routine and a parts placement diagram. The circuit diagram was not included but does come with the board.

After looking over the information, I was convinced, maybe, that software could handle the job. In fact, it almost looked too simple! So the next day I placed my order. Gimix said it would be shipped from stock.

Six days later, I had a package from Gimix in my hand. It had been shipped via UPS, blue label, which is air mail!

The board was packaged like an Egyptian pharaoh. Removing the outer box revealed another inside, and upon opening this one I found a conductive bag containing the video board in perfect condition.

Hardware

The board is double-sided fiberglass with through-hole plating. It also has a solder mask and all ICs are socketed. The board's +5 volt is supplied by two 5 volt regulators, and the −5 and −12 volt are zener-regulated.

Twenty-nine bypass capacitors are used on this board and, along with the crystal-controlled clock, provide a stable display. The board layout is OK with the character density and margin position controls located at the top. The video ouput is supplied with a 5 foot cable and connector.

Format

The board displays uppercase ASCII only, with a format of 16 lines of 32 or 64 characters per line. To change to 64 characters requires six easy trace cuts and the addition of six short jumpers. The software driver is also changed in six places.

The board contains 1K bytes of RAM, one location for each character position. This memory can be jumper addressed to any 1K section of memory. When using 32 character lines, the display does not need the upper

```
/
/ THIS ROUTINE DISPLAYS THE
/ CHARACTER CONTAINED BY ACCA
/ AT THE CURSOR LOCATION. IT
/ IGNORES ALL LOWER CASE CHAR-
/ ACTERS, AND ALL CONTROL CHAR-
/ ACTERS EXCEPT CR. IF THE CUR-
/ SOR MOVES OFF THE BOTTOM OF
/ THE SCREEN, ALL THE TEXT ON
/ THE SCREEN WILL BE SCROLLED
/ UP 1 LINE.
/
/ THIS ROUTINE IS FULLY RELO-
/ CATABLE, AND MAY BE PUT IN
/ PROTECTED MEMORY OR PROM.
/ IT DOES NOT AFFECT ACCA OR
/ ACCB, BUT WILL DESTROY IX.
/
/ MEMORY USAGE:
/ THE VIDEO BOARD HAS 1K OF
/ RAM WHICH THE ROUTINE ASSUMES
/ IS AT (HEX) DØØØ. DØØØ-D1FF
/ IS THE 32 X 16 DISPLAY. D2ØØ-
/ D3FF IS PRESENT REGARDLESS OF
/ THE DISPLAY SIZE.
/ THE CURSOR LOCATION IS STORED
/ IN 1C-1D (24-25).
/
/ THE ROUTINE USES 84 (54H)
/ BYTES OF MEMORY.
/
/ TO CONVERT TO 32 X 16,
/ USE THE LINES MARKED "/>"
/ TO REPLACE THE LINES
/ JUST BENEATH THEM.
/
/ SYMBOLS USED IN THIS ASSEM-
/ BLER:
/ "#": IMMEDIATE OPERAND
/ "*": TWO-BYTE IMMEDIATE
       OPERAND
/ "%": DIRECT ADDRESS
/ "X": INDEXED ADDRESS
/ "@": RELATIVE ADDRESS
/
/ ALL NUMERIC VALUES ARE GIVEN
/ IN HEXADECIMAL NOTATION
/
/ PREFACE ROUTINE
/ SAVES & RESTORES ACCB
P PSHB
  BSR  @A
  PULB
  RTS
/
/ TEST FOR CONTROL CHARS
A BITA #EØ
  BEQ  @C
/ TEST FOR BIT 8 = 1
  BMI  @Z
/ TEST FOR LOWER CASE LETTERS
  CMPA #6Ø
  BGE  @Z
/
/ DISPLAY CHARACTER
  LDX  %1C
  STAA XØØ
  INX
  STX  %1C
  BRA  @T
/
/ TEST CONTROL CHAR FOR CR
C CMPA #ØD
  BNE  @Z
/ CARRIAGE RETURN ROUTINE
  LDAB %1D
/>ANDB #EØ
  ANDB #CØ
/>ADDB #2Ø
  ADDB #4Ø
  STAB %1D
  LDAB %1C
  ADCB #ØØ
  STAB %1C
/ LOAD IX FOR TEST
  LDX  %1C
/
/ TEST FOR CURSOR OFF BOTTOM
/>T CPX  *D2ØØ
T CPX  *D4ØØ
  BNE  @Z
/
/ SCROLL TEXT UP 1 LINE
  LDX  *DØØØ
/>S LDAB X2Ø
S LDAB X4Ø
  STAB XØØ
  INX
/>CPX  *D1EØ
  CPX  *D3CØ
  BNE  @S
  STX  %1C
/
/ ERASE LEFTOVER TEXT
/>LDX  *D1FF
  LDX  *D3FF
  LDAB #'
E STAB XØØ
  CPX  %1C
  BEQ  @Z
  DEX
  BRA  @E
/
/ ROUTINE HOMES CURSOR
/ ROUTINE ADDRESS=(P)+4E
H LDX  *DØØØ
  STX  %1C
Z RTS
```

Listing 1. The Gimix video driver routine that comes with the board. Note the simplicity of each subroutine. (Repeated by permission of Gimix, Inc., 1337 W. 37th Place, Chicago IL 60609.)

512 bytes, which can be used for program storage or whatever. Accessing any of the video board's memory momentarily blanks the display. This causes a little black snow on the screen, but since most of the display is black, it is hardly noticeable.

The hardware takes care of displaying the characters stored in memory so all the driver program has to do is get a character from the A register and put it in the next memory location, keep track of the current cursor position and do the scrolling. Screen refresh is done by the hardware, not the processor.

Software

A simple, but adequate, display driver is included (see Listing 1). Routines to home cursor, erase from cursor to end of screen and scroll the display are provided. The driver is written for low memory using direct addressing but is relatively easy to assemble for another location. The direct addressing will have to be changed to relative addressing if the program is moved off the first page of memory.

If bit 8 of a display location is a one, a solid white block will be displayed regardless of the other bits. I modified the driver routine using this feature to display a white block at the current cursor position. This type of memory-mapped display allows the processor to update any display position quickly, making it useful for limited graphics as well as alphanumerics. The versatility is in the software. Gimix also sells a 2K ROM monitor that contains a 64 character driver routine.

The only time I slowed down was when I wanted to change the address of the display memory. It would have been helpful if a chart was included showing the A10-A15 jumpers to use for different 1K boundaries.

I have been pleased with the product and the service I received from Gimix. If you are looking for 6800 hardware, you might like to see what Gimix has to offer. I hear their new mainframe is built like a tank, and they are working on a new super video board. ■

Hashing It Out

With this scheme, you can save time and memory on your computer.

Jon A. Kapecki
161 Crosman Terrace
Rochester, NY 14620

One of the drudgeries of an interpretive language (from a computer's point of view) is that the machine must continuously repeat the same functions. Consider the simple loop in this program:

```
10 FOR I = 1 to 1000
20 LET V(I) = 0
30 NEXT I
```

The computer must decode those FOR, LET and NEXT keywords a thousand times, when all it really wants to do is zero that vector.

A typical BASIC interpreter can have 60 commands, functions and keywords to sift through, so decoding can become quite a job in itself. Also, the computer is not content to compare just the first character before deciding it hasn't found a match (for example, REM and RUN statements).

Solutions

An alternative method is to limit the system to single-letter commands (as in PILOT). However, this can result in software that is difficult to read and lacking self-documentation.

Another solution is to let the computer handle the abbreviating internally. Rather than store the full text of a multicharacter command, the interpreter stores an unambiguous "token," or single byte code, representing the command. (These kinds of interpreters are sometimes called "incremental compilers.")

This method saves memory and disk space and allows plenty of time on program entry to do the table look-up. LISTing the program requires retranslation of the token back into the character codes for the command, but this is usually a small price to pay. Most BASIC interpreters use a variant of the token technique.

Hashing, the Connoisseur's Delight

In the hashing system, the ASCII codes for the incoming keyword or command are combined by a mathematical process (see Table 1) to produce a single integer. A good hashing scheme at the machine-language level uses simple operations the computer can perform quickly. The system keywords (along with the corresponding jump addresses) have already been hashed by the same formula and placed in a table or tables. The interpreter scans the table for a simple byte match. For N keywords, an average of N/2 tries are needed for a match, if we ignore the frequency distribution of command words. (For obvious reasons, this process is called a linear search).

If we arrange the hashed entries in numerical order, we can perform a more efficient binary search. First, we must start in the middle of the table and determine whether the sought after hash code is larger or smaller than the midpoint value (if it's the same, you found your match). Then we go to the middle of the appropriate half and repeat the process. (You need a table of length $2^n - 1$, but you can fill out a short table with dummy entries or modify the algorithm slightly.)

A binary savings grows dramatically as the table gets bigger. For instance, a 64 keyword system takes an average of 32 tries for a linear search, as opposed to 6 for a binary search ($x = \log_2 n = \log_2 64 = 6$). The savings in search time may not be worth the extra code for short lists, however.

To add an item to a binary search table is cumbersome; you usually have to reorder the entire table. However, for software as fixed as an interpreter, that's usually of limited importance. A bigger problem is that more than one word can produce the same hash code. How often this happens depends on the size of the hashed number, the length of the keywords and the hashing algorithm. This is no problem for keywords that we get to choose, but a typing error or other random set of characters can unwittingly initiate a valid command. Careful choice of the hashing parameters can minimize these "collisions."

Application

Even though you're not planning to write an interpreter or modify one, hashing can still prove useful. For example, if you have a company of 100 or fewer employees, each of whom is tagged with an arbitrary insurance number from 1 to 20,000, you may want to write a program that will identify the employee's name with the in-

1. Rotate "A" twice left.	$= 1404_8$
2. Rotate "B" once left.	$= 604_8$
3. Add result of step 2 to step 1.	$= 2210_8$
4. Add "S" to the result of step 3.	$= 2533_8$

Thus, if L_n is the letter code to be hashed, the algorithm is $L_1 * 2^2 + L_2 * 2 + L_3$.

Table 1. An actual scheme used to hash three-letter keywords into a 12-bit code starting with the 8-bit ASCII for each letter. Only shifts and additions are used; the example is for the function ABS.

```
10 REM: HASH CODING DEMO -- EMPLOYEE INSURANCE NUMBER LOOK-UP
15 REM: J. A. KAPECKI -- JUNE '79  (INSURVER1.0)
20 DIM E%(100),A$(100)
30 REM: DATA ENTRY SECTION
40 FOR I=1 TO 100 \ LET E%(I)=0% \ NEXT I
50 L%=1%
55 PRINT 'ENTER DONE TO END DATA ENTRY'
60 IF L%>100 THEN PRINT '***** LIST FULL' \ GO TO 250
70 PRINT 'EMPLOYEE NAME ';
80 LINPUT N$
90 IF N$='DONE' THEN 250
100 PRINT 'INSURANCE #';
110 INPUT T
120 FOR I=1 TO 100 \ IF T=E%(I) THEN 540 \ NEXT I
140 GOSUB 450
150 IF F%=2% THEN 70
155 L%=L%+1%
160 LET J%=T-INT(T/100)*100+1
170 IF E%(J%)=0% THEN 210
180 LET J%=J%+1%
190 IF J%=101% THEN J%=1%
200 GO TO 170
210 LET E%(J%)=T
220 LET A$(J%)=N$
230 GO TO 60
240 REM: RETRIEVAL SECTION
250 PRINT  \ PRINT  \ PRINT
270 PRINT 'INSURANCE # (ENTER 0 TO STOP)';
280 INPUT T
290 IF T=0 THEN 570
300 GOSUB 450
310 IF F%=2% THEN 250
320 LET J%=T-INT(T/100)*100+1
325 LET D%=J%
330 IF E%(J%)=T THEN 400
340 LET J%=J%+1%
345 IF J%=D% THEN 420
350 IF J%<=100% THEN 330
370 LET J%=1%
390 GO TO 330
400 PRINT 'EMPLOYEE IS ';A$(J%)
410 GO TO 250
420 REM: EMPLOYEE NOT FOUND
430 PRINT '***** NO EMPLOYEE FOUND WITH INSURANCE #';T
440 GO TO 250
450 REM: VALIDITY CHECK
460 IF T>20000 THEN 500
470 IF T<=0 THEN 500
480 LET F%=1%
490 RETURN
500 PRINT '***** INVALID INSURANCE NUMBER'
510 PRINT '       MUST BE FROM 1 TO 20000'
520 LET F%=2%
530 RETURN
540 REM: DUPLICATE NUMBER CATCHER
550 PRINT '***** INSURANCE #';T;' ALREADY ASSIGNED'
560 GO TO 70
570 END
```

Listing 1. Demo program that implements a hashing scheme combined with a linear search to establish a list of insurance numbers with corresponding names and then retrieve the name for a given insurance number.

surance number.

You could execute a linear search of an insurance number list keyed to a name list, but as your lists became larger, searches would take increasingly longer. A binary search would be faster, but the frequent addition or deletion of names is awkward and usually would require adjustment of the search parameters to a new power of two.

A simple, yet fast, technique is to store the names in a string vector A$(N) so that calling up insurance number 8903, for example, would involve little more than PRINTing A$(8903). But this would require dimensioning A$ to 20000. Even if your version of BASIC allows this (some set upper limits to array subscripts) and you have the memory to do it, this simple approach wastes space. Only about ½ percent or less of the A$ vector would be populated (i.e., a "sparse vector").

With hashing techniques, we can reduce the storage requirements for our example to two vectors of 100 entries, about one percent of the space requirements of the simple approach. The saving can be even greater if you must explicitly dimension the length of your string vectors (as in Hewlett-Packard BASIC, for example, or those BASICs permitting virtual arrays).

First, we take the insurance number (T) and hash it so the result (J) falls between 1 and 100 (we've avoided 0 only because some BASICs won't allow it as a subscript). A simple algorithm for doing this is

$$J = T - \mathrm{INT}(T/100) * 100 + 1$$

Then we store the insurance number T as the Jth item in vector E(100) and the corresponding name as the Jth item in vector A$(100). To retrieve the name, we hash the insurance number as above and print A$(J). No searching!

Many insurance numbers can hash to the same vector location. To handle such collisions on entry, we first check to see if E(J) is empty (equal to zero). If it is, we can store T there. If not, we must find the next available empty slot and put T there along with the name in the corresponding position in A$.

On retrieval, we check to see if E(J) = T. If so, we print A$(J) as above. If not, we begin a linear search from that position until we find E(J) = T. The corresponding A$(J) is the one we want. To take care of overcrowding at the top of the table, we wrap it around to the first entry; that is, we create a "circular vector," so that if E(100) is occupied, we proceed to E(1).

The BASIC program in Listing 1 shows how such a scheme is implemented. The variables are the same as in the discussion above, except that a percentage sign (%) following a variable or constant designates an integer value (a space-saving feature in this interpreter). These signs can be eliminated with no changes in program execution.

To set a good example for routines that might be derived from it, the program also checks to see if the list is full (line 60), the insurance number is valid (line 450), the insurance number is not duplicated (line 120), and that the list is not traversed more than once looking for a nonexistent account (line 345).

Though you may never need an insurance number look-up, applications of these techniques to similar problems (serial numbers, record albums, car license numbers) involving sparse vectors are easy to do. ■

COMPUTER CLINIC

I am interested in obtaining information about existing computerized methods of editing and displaying music. I am not specifically interested in playing music on a computer, but am interested in the process of getting musical information into the computer, editing it, displaying it and printing or plotting it out.

Mike Firth
104 N. St. Mary
Dallas, TX 75214

I need schematics and operation manuals for the GTE Novar Selectric typewriters, series 5500 through 5570. These are no longer available from GTE or as separate items in the surplus market.

Joseph M. Kuc
5344 W. Winnemac
Chicago, IL 60630

We are attempting to locate a source for MICR readers (they read the strange-looking numbers at the bottom of checks) that will interface with an Apple II microcomputer. If anyone knows of such a source, please contact me directly.

E.C. Martin
President
Illinois Computer Mart, Inc.
1114 West Main
Carbondale, IL 62901

I would appreciate information from anyone that is currently uploading and downloading text or data files between Apples and a DEC 11/70 utilizing BASIC.

John E. Konopacky
Northeast Educational Processing Lab
1927 Main St.
Green Bay, WI 54301

Does anyone have any schematics or other information for a 1977 Imsai 32K dynamic RAM board?

Rusty Meadows
Box 169
Lake Dallas, TX 75065

We are a group of Apple II owners in Saudi Arabia that is interested in corresponding with clubs and individuals so that we can keep up to date on what is happening with microcomputers in general in the U.S. and the Apple in particular. We are also interested in swapping disks. Most of us have Apple IIs with two disks and the Pascal Language System.

C. Brandon Gresham, Jr.
Red Sea Apple Club
Saudi Arabian Parsons, Ltd.
PO Box 3694
Jeddah, Saudi Arabia

I am looking for information regarding the existence or planned formation of a user's group or club for the TI 99/4.

Larry Morrow
8075 Spring Garden Court
W. Chester, OH 45069

Since taking on the repair of my Interact home computer (model one), I have found that the company has gone out of business and has not released any information. I have not been able to find any service data. I have heard rumors that engineers from Interact gave information to a computer club somewhere in Michigan. If this is true, or if anyone knows where I may find the data I need, please contact me.

Stephen Carrel
RCA Solid State Division
Route 12
Findlay, OH 45840

I am running Microsoft BASIC in ROM supplied by Netronics that will not execute certain functions the first time they are used in a program (e.g., CHR$ and others). The problem seems only to be associated with string operations and invariably disappears the second time the statement is executed in the program. My memory passes every test, and Netronics tells me that the ROM works in their shop computer. Can anyone help me with this problem?

Colin Evans
150 Walnut St.
Stratford, CT 06497

I recently acquired an ITEL 1051 Model #78-10-10-10 computer terminal manufactured by Dura International of Greeley, CO (now out of business), in about 1970–72. After many phone calls and many "we are not interested" answers, I was finally able to acquire the schematic logic diagram. I was not able to find an operations manual. I am not proficient enough at engineering to dope out all of the machine's functions. Could someone provide me with a photocopy of the operator's manual?

Also, I have an Elf II by Netronics using 4K memory, the Giant Board and video board. I need a program that is not attached to BASIC in order to drive the RS-232-C I/O, so that I may use the RS-232-C for some of my machine-language programs.

James Wicks
1970-A Cedar Ave.
Long Beach, CA 90806

CLUB NOTES

Woodsbridge, VA

The Prince William Computer Club holds its regular meetings at the Prince William Branch Library in Woodsbridge, VA, on the first Tuesday of each month at 7:30 PM. For information, call Don Bennett, 703-670-4773.

Cedar Rapids, IA

A PET user's group is active in the Cedar Rapids, IA, area under the direction of Don Vorhies, chairman. Write to Don at 1321 42 St., SE, Cedar Rapids, IA 52403, for more information.

Salem, OR

The Salem Area Computer Club meets on the first Monday of each month at the McKinley Community School on McGilchrist Street in Salem on odd-numbered months and at the Computer Pathways Unlimited Retail Store in the Lancaster Mall in Salem on the even-numbered months. Club membership—$5 per year—covers the cost of the club's monthly newsletter. For further information, contact Doug Walker, 3485 Mock Orange Ct., S., Salem, OR 97302, 503-364-2488.

Hamilton, Ontario

The OSI User's Group of Southern Ontario has released the following meeting dates for the 1980/81 schedule: Dec. 6, March 7, June 6 and Sept. 5. For more information, contact Dr. N. Solntseff or C. Bryce, Unit for Computer Science, McMaster University, Hamilton, Ontario L8S 4K1, 416-525-9140, ext. 4680 or 2065.

Ann Arbor, MI

OSI-MUG—the Ohio Scientific Michigan User's Group—has been in operation since May and has an initial membership of approximately 130 people from primarily the southeastern Michigan area. For further information contact Ralph V. Johnson, Sr., 3247 Lakewood Ave., Ann Arbor, MI 48103 (313-761-5358).

Dear Subscriber:

Kilobaud Microcomputing does not keep subscription records on the premises. Therefore, calling the Peterborough offices doesn't solve your subscription problem.

To quickly solve your problem, please send your most recent address label and a description of the problem to:

Kilobaud Microcomputing
Subscription Department
PO Box 997
Farmingdale, NY 11737

Please allow the subscription department at least two weeks for an answer or a solution to your problem.

Thank you and enjoy your subscription.

Sincerely,

Debra L. Boudreau

Debra L. Boudrieau
Circulation Manager

Edited by Eric Maloney

MICRO-SCOPE

Astrology Goes Computer

The next time you go have your astrology chart read, a microcomputer may be doing most of the work.

"The computer revolution has just started for us, and we'll soon be in the middle of it," National Astrological Society director Barbara Somerfield told the Associated Press at the Society's national conference in late August. "Last year only about 1 percent of all astrologers had computers to help in their calculations. This year it's 5 percent, next year 20 percent and in a few years everyone will have one."

Astrologers are using micros primarily to eliminate much of the mathematical drudge work involved in erecting natal (birth) charts. An astrologer can often spend hours doing all sorts of logarithmic calculations to determine the planets' positions at the time of birth. A micro can do it in a matter of seconds.

"It gives us more time to concentrate on the essential part of our craft—interpreting the data and helping a person realize their full potential," says Somerfield.

In addition, some astrologers are writing their own programs to compile statistics for research or for special functions. Charles A. Jayne, an owner of two Commodore PETs, describes in an article in *Astrology Plus* several of these programs. One, called TRISHIFT, can shift all of the planets in an individual's horoscope to any locality in the world, to determine the compatibility of that person to that particular place. Such computations, if done by hand, could take days.

Already, several computer astrologers are making their marks in the world of astrology. Michael Erlewine of Big Rapids, MI, is writing programs for the TRS-80, Apple and PET, and markets these and other people's programs through his own organization. The American Federation of Astrologers also sells his programs, and will soon be publishing a series of books on microcomputing for the astrologer.

"One can truly say that the mechanical and technical side of astrology is now already being revolutionized, and that this ought to be beneficial in its effects," Jayne concludes.

"My Apple's a Scorpio. What's yours?"

Ashes to Ashes, Dust to Disk

If you live in the Pacific Northwest and haven't heard yet—or if you've got an active volcano in your backyard—be warned: volcanic ash is not healthy for your computer.

Lewis A. Whitaker, executive vice-president of Innovative Computer Products in Tarzana, CA, suggests that computerists in the fallout area take the following steps to minimize damage and data loss:

Keep magnetic media in covered containers. Disk cartridges, cassettes, magnetic tape and floppy disks may look hardy, but they are extremely vulnerable to microscopic dirt particles.

Cover equipment when not in use. Use a plastic typewriter-type cover to cover the disk drives, printer and CRT. It is better to keep contamination from a computer surface than to try and remove it once a problem has occurred.

Periodically maintain media. Cleaning and testing of magnetic media will not only lengthen the life of media, but will help maintain error-free processing over the life of the media.

Maintain drives frequently. Dirt seems to gravitate to heads of magnetic media equipment. While hard disk drives do not have a head-to-media contact and, therefore, do not need to be cleaned as frequently, magnetic tape, cassette tape and diskettes all have head-to-media contact, and cleaning of these heads on a daily basis would surely minimize contamination-caused problems.

Computer Blamed for Massacre

A computer has been blamed for what one scientist at the University of California called "the Guyana massacre of mice."

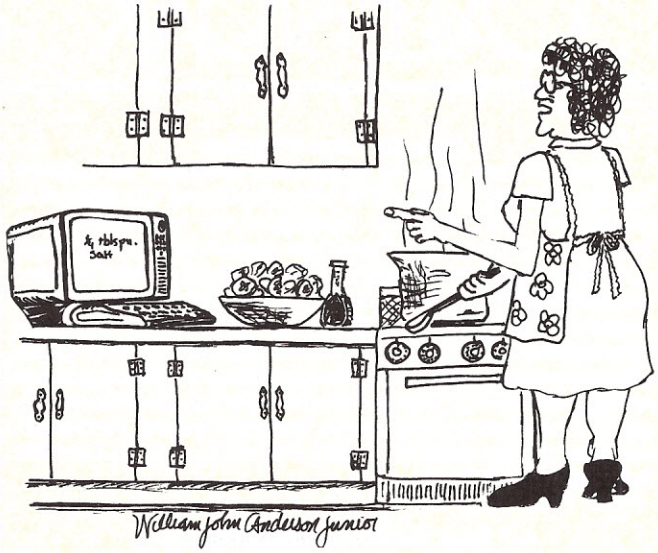
Some 1500 of the rodents, part of a $1 million biological research project, turned belly-up last August when a computer designed to control the temperature in their storage area malfunctioned, says an article in the Washington *Post*. The temperature rose to 100 degrees, leaving only 500 survivors.

The article did not say whether the computer has been turned over to the local humane society.

Cooking with CompuServe

News and stews from *Better Homes and Gardens* magazine are now available to CompuServe information service subscribers.

Information related to the magazine's monthly recipe features will be offered with detailed nutritional and calorie analyses. Other features being planned include complete menus built



"Quit picking at the salad!"

around a recipe, approximate food costs per serving and recipes in addition to those appearing in the magazine.

The service will be expanded to cover other areas, such as gardening, building, decorating, crafts, travel and money management.

CompuServe subscribers will have access to some of the raw material used by the magazine's editors that is not included in the final publication of the monthly magazine.

Games "On Way Out"?

Computer games are "on the way out," says a man who has spent the last three years inventing them.

Joseph Willhide, creator of the Mathemagician teaching calculator, told the Boston *Globe* recently that the market is saturated and will soon experience a "shakeout." The trend, he says, is toward electronic toys, where there are more opportunities for creativity and innovation.

"The consumer does not spend that much time choosing a game, and it just becomes tougher and tougher to come up with things that are perceived as new by the public," he says.

Willhide adds that the games industry will have to adjust when the consumer learns how to evaluate games and can determine whether he is getting his money's worth.

Journals Selling More Ads

Display advertising was up 7 percent and ad revenues up 19.4 percent in 16 computer and data communications journals during the first half of 1980, compared with the same period in 1979, a C System study shows.

Computerworld enjoyed the largest increase in total ad pages, up 226 pages (to 2368 pages) for a gain of more than 10 percent. *Computer Systems News* had the second largest increase, up 156 pages for a gain of more than 50 percent. *Computer Design* and *Mini-Micro Systems* each showed nearly a 150-page increase over the first half of 1979. C Systems Ltd. specializes in computerized analyses of trade journal and business publication advertising activity.

CALENDAR

Computer Crime Workshops

Computer Crime Info, an international conference on computer security and fraud control, will be held at the Crystal City Marriott in Washington, D.C.,' Dec. 1-3. Participants will include Joseph E. Henehan, chief of the White Collar Crime Section of the Federal Bureau of Investigation; Robert P. Campbell, president of Advanced Information Management, Inc., and general chairman of Computer Crime Info; Robert V. Head, federal executive fellow of the Brookings Institution; John Michael Williams, director of information security of the System Development Corporation; Carl Hammer, director of computer sciences at Sperry Univac; J. T. Westermeier, attorney at law; and PJ Corum, director of Computer Auditing Systems, Pansophic Systems. For information write the Information Exchange, 1730 North Lynn St., Suite 400, Arlington, VA 22209.

Intro, Troubleshooting Workshops

Integrated Computer Systems has set its winter schedule for its Hands-On Microprocessor workshop. The workshop is set for Dec. 2-5 in Chicago, Dec. 9-12 in Cherry Hill (Philadelphia), Jan. 13-16 in San Diego and Jan. 27-30 in Washington, D.C. Its Hands-on Microprocessor Troubleshooting workshop is scheduled for Dec. 2-5 in Sunnyvale, Dec. 9-12 in Cherry Hill, Jan. 20-23 in San Diego, and Feb. 3-6 in Washington, D.C. For more information, contact Ruth Dordick at Integrated Computer Systems, 3304 Pico Blvd., PO Box 5339, Santa Monica, CA 90405 (213-450-2060).

Oklahoma Workshops

Oklahoma State University at Stillwater has two workshops scheduled this fall and winter. "Microcomputer Workshop," an introduction to microcomputers, is set for Oct. 20-21 and Dec. 4-5. "Microcomputer Systems and Interfacing," a program for persons with little experience who are using or maintaining microcomputer systems, is scheduled for Oct. 20-21. For further information contact Technology Extension, 313 Crutchfield, Stillwater, OK 74078 (405-624-5714).

Virginia Tech Workshops

Four workshops are set at the Virginia Tech campus in Blacksburg, VA. The programs will be directed by Dr. Paul Field, Dr. Chris Titus, Dr. Jon Titus, Mr. Andy Staugaard and Mr. David Larsen. The workshops are "Digital Electronics for Automation and Instrumentation," Dec. 8, 9 and 10; "Microcomputer Interfacing Programming and Application Using the 280/8085/8080," Dec. 11, 12 and 13; "TRS-80 Radio Shack Microcomputer Interfacing and Programming for Scientific Instrumentation," Dec. 15, 16 and 17; and "Motorola Single Chip Interfacing and Programming Using the 6801, 6809 and 6800," Dec. 18, 19 and 20.

For more information, contact Dr. Linda Leffel, C.E.C., Virginia Tech, Blacksburg, VA 24061 (703-961-5241).

Arizona Microcomputer Conference

The College of Education at Arizona State University, Tempe, AZ, will host a special microcomputer conference Jan. 16-17, designed to introduce educators to the many applications of microcomputers in the classroom.

The goal is to provide an awareness of microcomputers and their impact on society and ways that microcomputers are currently being used in education at the elementary and secondary levels, in the fine arts areas, in career and vocational education, and in special education. Dr. Gary G. Bitter, Arizona State University, Payne 203, Tempe, AZ 85281.

New Mexico Computer Fair

The New Mexico Computer Society is hosting the second annual New Mexico Computer Fair at the Civic Auditorium in Albuquerque, NM, Nov. 15 from 10 AM until 8 PM. Admission is free. For more information contact Ron Benninghoff at 505-831-3683 or 505-836-0065 after 4 PM.

PET to NEC and CENTRONICS PRINTER ADAPTER

LOWEST COST COMPLETE INTERFACE ON THE MARKET

Simple to use—low cost—designed for NEC 5530 Spinwriter and Centronics parallel printers. Works with WORDPRO and other software. Switch for upper-lower case conversion or upper case only.

Plugs into the PET and into the printer—all cables and connectors included—extra IEEE connector for Commodore disk drives.

Uses BASIC PRINT statements—no machine code needed.

Device address selectable—works with other peripherals.

$129 complete—compare to others at $225. Generous dealer discounts.

Assembled and tested. Our usual 30 day money back trial period applies.

Order direct or contact your local computer store.

CmC **Connecticut microComputer, Inc.** ✓182
34 Del Mar Drive, Brookfield, CT 06804
(203) 775-4595 TWX 710 456-0052

VISA and M/C accepted—send account number, expiration date and sign order. Add $3 per order for shipping & handling —Foreign orders add 10% for air postage.
Mention this magazine with your order and deduct 2%.

779 UPPER CASE/lower case "Conversion Kit I"

Expand the capabilities of your 779 line printer to include word processing!! Available to all Centronics 779 and TRS 80 Printer I owners is the option of lower case and changing slash 0 Zero to standard 0. No etch cuts or soldering needed. Installs in minutes with a screwdriver. No program modification or additional interface is required. **Price $125.00**

UPPER/LOWER CASE NOW AVAILABLE FOR THE FOLLOWING CENTRONICS PRINTERS:
101AL, 102BL, 306, 500, 501, 503, 700, 701, 702, 703, 780, 781.

Motor Control "CONVERSION KIT II"

FOR ALL CENTRONICS 779 & TRS 80 PRINTER I LINE PRINTERS!!

Our "Conversion Kit II" Motor Controller gives your 779 the ability to turn the motor on and off automatically. Removes the annoying noise of constant run, increasing the life span of your 779 and TRS 80 line printer motor! No soldering, software or hardware changes needed. Installs easily. **Price $95.00**

SAVE! Buy Service Technologies "Conversion Kit I" and "Conversion Kit II" together for the single price of **$199.00**

To order, please send check or money order in the proper amount to:
Service Technologies, Inc.
32 Nightingale Rd. ✓208
Nashua, N.H. 03062
(603) 883-5369

Visa and Master Charge accepted (please include signature, expiration date and phone number).

6502	7.45	10/6.95	50/ 6.55	100/ 6.15
6502A	8.40	10/ 7.95	50/ 7.35	100/ 6.90
6520 PIA	5.15	10/ 4.90	50/ 4.45	100/ 4.15
6522 VIA	6.90	10/ 6.50	50/ 6.10	100/ 5.70
6532	7.90	10/ 7.40	50/ 7.00	100/ 6.60
2114-L450		4.45	20/ 4.25	100/ 3.95
2114-L300		5.65	20/ 5.35	100/ 4.95
2716 EPROM (5 volt)		13.45	5 /12.75	10/11.85
2732 EPROM (5 volt)				42.00
4116-200 ns RAM (NEC)				**8 for 41.90**
6550 RAM (PET 8K)				12.70
S-100 Wire Wrap $2.65			Solder Tail	$2.15

DISKS

(write for quantity prices)

SCOTCH (3M) 8"	10/3.00	50/2.85	100/2.75
SCOTCH (3M) 5"	10/2.95	50/2.80	100/2.70
Maxell 5"	10/3.65	50/3.40	100/3.15
Maxell 8" Double Dens.	10/4.10	50/3.95	100/3.80
Verbatim 5"	10/2.40	50/2.35	100/2.30
(add 1.00 for plastic storage box)			
BASF 5" soft	10/2.40	20/2.35	100/2.30
BASF 8" soft	10/2.40	20/2.35	100/2.30
Diskette Storage Pages			10 for 3.95
Disk Library Cases		8"-2.85	5"-2.15

A P Products 15% OFF

A P HOBBY-BLOX 15% OFF

All books 15% off

KIM-1	$159
SYM-1	$209
KTM 2/80	$349
Leedex Monitor	$129
Centronics 737	$800
C-10 Cassettes (AGFA PE611)	10/5.65
NEC Spinwriter-parallel	$2450
XYMEC HI-Q 1000	$2150
Zenith Z19 Terminal	$739
Zenith Z89 with 48K	$2150
FORTH+ for PET or APPLE (full FIG version)	$60
PASCAL for PET	$75
Z80A 64K +2.4 Meg Disk	$3500

COMMODORE PET-CBM
Write or call for quotes
NEW 8016/32 80 column screen
NEW 8050 950K Dual Drive
EDUCATIONAL DISCOUNTS AVAILABLE

ATARI 800 $777 All Atari Modules 25% OFF

SPECIAL-purchase ATARI 800, receive extra 8K memory FREE.

EDUCATIONAL PLAN - buy 2 ATARI Computers, receive 1 ATARI **FREE!**

WRITE FOR CATALOG
Add $1.25 per prepaid order for US shipping (UPS)

✓121
A B Computers 115 E. Stump Road
Montgomeryville, PA 18936
(215) 699-5826

TOLL FREE ORDERING

These Fine Products and More

NORTHSTAR
HRZ 1-32K-D 2100
HRZ 2-32K-D 2340
HRZ 2-32K-O 2690
HARD DISC SYSTEM 3950

DYNA BYTE
DB 8/1 4 8K 2395
DB 8/2 4 8K 3900
DB 8/4 3030
32M PHOENIX 11800

TERMINALS
TELEVIDEO 912 745
TELEVIDEO 920 795
SOROC IQ-120 700

PRINTERS
NEC 5510 2700
NEC 5520 2975
TI-820 1650
ANADEX 795
BASE 2 600
EPSON . . . . CALL

THINKER TOYS
DISCUS 2 + 2 1 DRIVE 1265
DISCUS 2D 1 DRIVE 970
DISCUS M26 HARD DISC . . .4095

SOLID STATE MUSIC Kit Assm
SB1 SYNTHESIZER . . .161 227
VB1B VIDEO 125 170
CB2 Z80 CPU 168 220

MEASUREMENT SYSTEMS
DM 3200 32K 250ns 500
DM 6400 64K 250 ns 640
DMB 3200 650

SOFTWARE-DISCS-MISC
CPM-2 150
WORDSTAR 350
GRAHAM-DORIAN . . . . . . .CALL
STRUCTURED SYSTEMS . . CALL
VERBATIM 5 (10) 28
VERBATIM 8 (10) 35
ATARI . . . . CALL
TI 99-4 . . . . CALL

WE WILL TRY TO BEAT ANY ADVERTISED PRICE
Automated Equipment Inc.
4341 W. Commonwealth Ave Suite D
Fullerton, Calif. 92633 ✓96
(714) 739-4701 (800) 854-6003

CORRECTIONS

The parts list accompanying Fig. 1 of September's "Get Your PET on the IEEE 488 Bus" (part 3, p. 53) was inadvertently not published with the article. It is listed below. The parts numbers are in the hexagons in the original figure.

Number	Part	Vendor	Description
1, 15	CD 4049A	Motorola, RCA	CMOS hex inverter
2, 3, 4	7400	TI, National	TTL 4-2 NAND
5	7410	TI, National	TTL 3-3 NAND
6	9324	Fairchild	TTL five-bit comparator
7	7425	TI, National	TTL 2-4 NOR
8, 9, 10	MC 3448A	Motorola	GPIB interface
11	7402	TI, National	TTL 4-2 NOR
12	AY-3-1015	General Instruments	UART
13	7408	TI, Fairchild	TTL 4-2 AND
14	206-7	James Electronics	Seven-position DIP switch
16			10k resistors, 5 percent

A section of Fig. 1 in Dexter French's "A Hardware Calendar Clock for Your 6800" (June 1980) was incorrect as published. The corrected section of the schematic appears in Fig. 1.

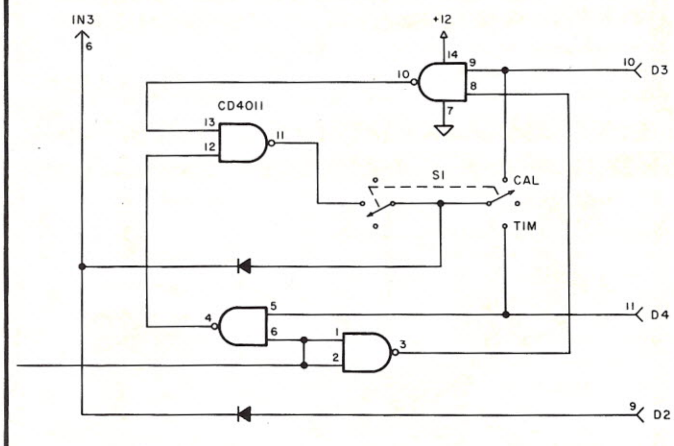


Fig. 1.

The price for the PTS-3 interface unit was incorrectly listed in the August 1980 *Microcomputing* New Products section (p. 15). It should have been $89.95, not $69.95.

"Disassembler for the 1802" (July 1980, p. 196) contains two structural errors. The op code IRX ($60) outputs as OUT 0, and the invalid op code $68 outputs as INP 0. The patches needed to correct these errors, so $60 disassembles as IRX and $68 as INVALID, are shown below. Note that the user space now starts at $07DE, instead of $07C0.

```
04A7— C0 07 C0

07C0— 52 32 D4 FF 08 CA 04 AA
07C8— 79 DB 49 4E 56 41 4C 49
07D0— 44 00 30 DA 79 DB 49 52
07D8— 58 00 22 C0 04 3C
```

David Henderson
Puyallup, WA 98371

CLASSIFIEDS

Classified advertisements are intended for use by persons desiring to buy, sell or trade used computer equipment. No commercial ads are accepted.

Two sizes of ads are available. The $5 box allows up to 5 lines of about 35 characters per line, including spaces and punctuation. The $10 box allows up to 10 lines. Minimize use of capital letters to save space. No special layouts allowed. Payment is required in advance with ad copy. We cannot bill or accept credit.

Advertising text and payment must reach us 60 days in advance of publication (i.e., copy for March issue, mailed in February, must be here by Jan. 1). The publisher reserves the right to refuse questionable or inapplicable advertisements. Mail copy with payment to: **Classifieds, Kilobaud Microcomputing, Peterborough NH 03458.** Do not include any other material with your ad as it may be delayed.

Digital Group System Z-80 CPU, 26K memory, 16×64 CRT, 2 Phi-Decks, Expandor printer, extensive software, all documentation and loads of extras. $2000 negotiable. Paul. A. Teseny, 17 DeKalb Place, Morristown NJ 07960. 201-539-6876, evenings.

For Sale: New 16K RAMs from scrapped computer boards. Tested, 200 ns 4116s. High rel. ceramic. $45 for 8, $80 for 16. D. Gennetten, 4425 Goshawk Drive, Ft. Collins, CO 80526. 303-226-1395.

For Sale: SYM-1 computer, $110, 4K RAM. In original box with all manuals. COD ok. Doug Gennetten, 4425 Goshawk Drive, Fort Collins, CO 80526. 303-226-1395.

For Sale: Digital Group Z-80 microcomputer with 42K RAM, keyboard, monitor, 2 digital tape decks and much software. Complete $1300. Phone 805-968-5893 or write: Steven Fornell, 7536 Evergreen, Goleta, CA 93017.

For Sale: 32K Apple II with RS-232, Suprmod, recorder and $125 in books and software. A $1500 value for $1275. Bob Biagioni, 525 Ahrens, Lombard, IL 60148. 312-627-6374.

For Sale: LSI-11/2 with 64KB memory, Andromeda FDC11 floppy controller (RX-01 compatible) with dual PerSci 277 drives, console and line printer interfaces, Heath box and power supply. $4595 or best offer. John Sterne, 3880 San Rafael Avenue, Los Angeles, CA 90065. 213-225-2474.

For Sale: H9 video terminal assem., $200. SYM-1 micro w/case, power, new monitor installed & 5 issues Micro, $225. G. Zimmerman, 113 8th St., Downers Grove, IL 60515. 312-969-1595.

Execuport portable terminal—300 baud—built in acoustic coupler. Can also be used as printer. With carrying case—$600 + shipping. 203-563-5750, after 9:00 PM.

Compucolor II microcomputer with 16K memory and built-in disk drive for sale. 1 year old $1500, 7620 Pagent Lane, Wichita, KS 67206, 316-684-4256.

For Sale: TI-59 Programmable Calculator with PC-100C printer cradle; both in excellent condition. Includes extra paper for printer, all manuals, my own library of programs. $380 or best offer—I pay shipping. Mike Smith, 908 Murray Hill Rd., Binghamton, NY 13903.

For Sale: Printer—Texas Instruments Silent 700 Electronic Data Terminal. Excellent, like-new condition. Can be interfaced with TRS-80 or others. $495. Call Debi 212-224-2448 (eve.).

For Sale: 4K static memory boards by Atwood Enterprises; four each with motherboard. A $320+ value for $200. Also, one each Ramsey TH3216 video board. $200+ value for $130. Want, TRS-80 expansion interface after selling above. M. Schuldt, 412 Donner Ave., Petaluma, CA 94952. 707-762-6975. Fine condition, I pay shipping.

For Sale—Mits: Two 16MCD RAM @ $200; four 4MCD RAM @ $50; ACR, $75. Clare-Pender ASCII kybd, $25. TTY ASR33, stand, long-roll paper, RS-232; $300 + shipping. DEC LA36, tray, table, RS-232; $750 + shipping. Moduperf 30 cps paper tape punch, interface plans included; $100. All working when last used. Documentation. Offers considered, but first check for stated price accepted. Details: John Neville, Box 400, Onamia, MN 56359. 612-532-3103.

For Sale: SYM-1 microcomputer with 8K RAM, BASIC in ROM (8K Microsoft), and enclosure, $295. KTM-2/40, $240. John Maslowski, 108 Meadowcrest Dr., Nanticoke, PA 18634. 717-735-2341.

For Sale: Integrand S-100 mainframe (non disk chassis) with P.S. ± 16 V, 3 A, + 8 V, 25 A, − 8 V, 3 A E.M.L. filter, active bus term, etc. 10 conn on 12 position M.B., all documentation & like new cond, $200. For Sale: Z-80 S.D. Starter System's complete microcomputer on-board hex display. Complete with 5 V P.S. & 25 V EPROM P.S. extra 1K mem, all doc., up and running, $150. Call 213-780-9378 after Oct 28th.

For Sale: Netronics ELF II in steel enclosure; 2—4K static RAM bds; Giant bd; kluge bd; Math/ROM bd; tape cntrl bd; rf modulator; video display bd & ASCII kybd in steel enclosure; and expansion pwr supply. All for $700. Netronics software, courses & books. Worth $115. Incl. free. Money order preferred. Write J. Absetz, Box 696, USNSGA, FPO New York, NY 09518.

Wanted: Synertek KTM-2 keyboard and/or Trendcom 200 printer. In return I will fabricate an equal value of prototype PC boards. R. Hegel, 7332 Portland Ave., Richfield, MN 55423.

Diablo Hytype 1 Model 1200. Best of the "daisy wheel" printers. Brand new units w/pin feed friction platen & print wheel. Interface for Apple, TRS-80 & CP/M systems, maintenance manual and additional interface info available. There is no better buy anywhere. After 6 PM. Scott Priester, 211 White Water Ct., Greer, SC 29651, 803-268-0678.

MICRO QUIZ

From page 11.

Answer: (0,0), (0,1) and (1,1).

A	B	$\overline{A+B}$	$\overline{(A+B)}+B$
0	0	1	1
0	1	0	1
1	0	0	0
1	1	0	1

Now NRI takes you inside the world's most popular microcomputer to train you at home as the new breed of computer specialist!

NRI teams up with Radio Shack to teach you how to use, program and service microcomputers...make you the complete technician.

It's no longer enough to be just a programmer or a technician. With microcomputers moving into the fabric of our lives (over 200,000 of the TRS-80™ alone have been sold), interdisciplinary skills are demanded. And NRI can prepare you with the first course of its kind, covering the complete world of the microcomputer.

Learn At Home in Your Spare Time

With NRI training, the programmer gains practical knowledge of hardware, enabling him to design simpler, more effective programs. And, with advanced programming skills, the technician can test and debug systems quickly and easily.

Only NRI gives you both kinds of training with the convenience of home study. No classroom pressures, no night school, no gasoline wasted. You learn at your convenience, at your own pace. Yet you're always backed by the NRI staff and your instructor, answering questions, giving you guidance, and helping you over the tough spots.

Training includes TRS-80 computer, transistorized volt-ohm meter, digital frequency counter, and the NRI Discovery Lab with hundreds of tests and experiments.

(TRS-80 is a trademark of the Radio Shack division of Tandy Corp.)

Explore the TRS-80 Inside and Out

NRI training is hands-on training, with practical experiments and demonstrations as the very foundation of your knowledge. You don't just program your computer, you introduce and correct faults ...watch how circuits interact...interface with other systems...gain a real insight into its nature.

You also build test instruments and the NRI Discovery Lab, performing over 60 separate experiments in the process. You learn how your trouble-shooting tools work, and gain greater understanding of the information they give you. Both microcomputer and equipment come as part of your training for you to use and keep.

Send for Free Catalog... No Salesman Will Call

Get all the details on this exciting course in NRI's free, 100-page catalog. It shows all equipment, lesson outlines, and facts on other electronics courses such as Complete Communications with CB, TV and Audio, Digital Electronics, and more. Send today, no salesman will ever bother you. Keep up with the latest technology as you learn on the world's most popular computer. If coupon has been used, write to NRI Schools, 3939 Wisconsin Ave., Washington, D.C. 20016.

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016
NO SALESMAN WILL CALL
Please check for one free catalog only.

All career courses approved under GI Bill.
☐ Check for details.

- ☐ Computer Electronics Including Microcomputers
- ☐ TV/Audio/Video Systems Servicing
- ☐ Complete Communications Electronics with CB • FCC Licenses • Aircraft, Mobile, Marine Electronics
- ☐ CB Specialists Course
- ☐ Digital Electronics • Electronic Technology • Basic Electronics
- ☐ Small Engine Repair
- ☐ Electrical Appliance Servicing
- ☐ Automotive Mechanics
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Refrigeration, & Heating including Solar Technology

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council 172-110

DEALER DIRECTORY

El Monte, CA

Ohio Scientific specialist in the San Gabriel valley serving greater Los Angeles. Full product line on display. Specializing in business computers. In-house service. Custom programming. Terminals. Printers. Open Mon–Sat, 9 AM–7 PM. **Computer & Video, 3380 Flair Dr., Suite 207, El Monte, CA 91731, 572-7292.**

N. Hollywood, CA

Wholesale prices to dealers & computer club members! Anadex, Atari, Base-2, Centronics, Emako, Godbout, Hazeltine, Lobo, Micropolis, MicroPro, NEC, Okidata, Paper Tiger, Soroc, Tarbell, Televideo, TI, Vector Graphic, Zenith & others. **Patio Computer Sales Co., 5437 Laurel Canyon Bl., #208, N. Hollywood, CA 91607, 762-0020.**

San Jose, CA

Bay area's newest computer store. Featuring the new Texas Instruments TI 99/4 home & business computer. Software for TRS-80, Apple, PET, etc. Magazines. **Hobbi-Tronics, 1378 S. Bascom Ave., San Jose, CA 95128, 998-1103.**

Santa Barbara, CA

Complete computer systems for business and personal use. Classes, seminars, word processing supplies, books, magazines. **Computers Plus, 1827 State St., Santa Barbara, CA 93101, 963-4542.**

Sarasota, FL

Dynabyte computer systems, Hazeltine and NEC, Word-Star, Structured Systems accounting. Consulting, training, sales, service. **Glisco, Inc., 4001 Roberts Point Rd., Sarasota, FL 33581, 349-0200.**

Tampa, FL

Apple Computer sales and service. S-100 boards from SSM, Godbout, Thinker Toys, California Computer Systems. Computer books and magazines. **AMF Microcomputer Center, Inc., 11158 N. 30th Street, Tampa, FL 33612, 971-4072, 977-0708.**

Aurora, IL

Microcomputer systems for home or business; peripherals, software, books & magazines. Apple, Hewlett-Packard, North Star, Cromemco systems. IDS-440G printer w/Apple graphics, New HP-85 & HP calculators. **Farnsworth Computer Center, 1891 N. Farnsworth Ave., Aurora, IL 60505, 851-3888.**

Chicago, IL

Computer Hardware Specialists for home and business. Largest selection of computer books, magazines and copyrighted software in Illinois. Experienced factory-trained service department. Feature Apple, Alpha Microsystems and Hewlett-Packard calculators and accessories. **Data Domain of Schaumburg, 1612 E. Algonquin Rd., Schaumburg, IL 60195, 397-8700.**

Chicago, IL

Brand new lowest prices, never undersold, postpaid in USA—Teletype 43 keyboard printers, Okidata & Integral Data printers, SS-50 bus computers, peripherals & business software. **Data Mart, 914 East Waverly Street, Arlington Heights, IL 60004, 398-8525.**

Garden City, MI

Complete systems for business, professional and personal applications. Custom programming available. Apple II, North Star, Vector Graphic and other lines of microcomputers, software, books, components. **Computer Center, 28251 Ford Rd., Garden City, MI 48135, 422-2570.**

Westland, MI

Integrated circuits, TTL, CMOS, linear. Many hard to find "S" and "LS" types. Resistors, capacitors, diodes, IC sockets and many other items. **Westland Electronics, 34245 Ford Rd., Westland, MI 48185, 728-0650.**

Dealers: Listings are **$15** per month in prepaid quarterly payments, or one yearly payment of **$150** also prepaid. Ads include 25 words describing your products and services plus your company name, address and phone. (No area codes or merchandise prices, please.) Call Marcia at 603-924-7138 or write *Kilobaud Microcomputing*, Ad Department, Peterborough, NH 03458.

Hannibal, MO

Ohio Scientific products, modifications, service, software. 8" disk for C1p, C4p. Process control specialist. **E&I Technical Service, 5300 Paris Gravel Road, Hannibal, MO 63401, 248-0084.**

St. Louis, MO

Experimenters' Paradise. Electronic and mechanical components. Computer People, Audio People, Hams, Robot Builders, Experimenters. Open six days a week. **Gateway Electronics Corp., 8123-25 Page Blvd., St. Louis, MO 63130, 427-6116.**

Portland, OR

Ohio Scientific specialists for business and personal computers. Local service. Terminals, printers, custom programming. Full OSI product line on display! 10 AM to 6 PM M–F. **Fial Computer, 11266 SE 21st Ave., Milwaukie, OR 97222, 654-9574.**

Milwaukee, WI

Specializing in the CBM-PET, business, personal, educational, industrial, telecomputing systems. Consulting, modems, printers, books, accessories, magazines, supplies, peripherals, timesharing. Factory authorized service. Convenient freeway access. **PETTED micro systems, 4265 W. Loomis Rd., Milwaukee, WI 53221, 282-4181.**

16K MEMORY KITS
TMS—4116 250 N.S.

FOR: APPLE, TRS-80, HEATH EXPANDORAM I & II AND MANY MORE

12 MONTH GUARANTEE

$48.00 PER SET OF 8 CHIPS

ADD $2.00 SHIPPING & HANDLING PER ORDER

PAYMENT: MASTER CHARGE, VISA CASH, MONEY ORDER, U.P.S., C.O.D., PERSONAL CHECKS REQUIRE 2-3 WKS. TO CLEAR BANK.

Computer Services ✓80

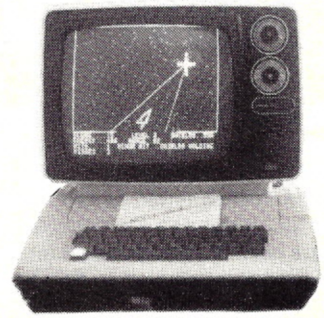
30 Hwy. 321, N.W.
P. O. Box 2292
Hickory, N. C. 28601
(704) 294-1616

PHONE HOURS:
6 P.M. — 9 P.M. EST

APPLE — JACK

the graphics & games people

New!
Super Starbase Gunner



$19.95 DISK **$19.95 DISK**

Most shoot-em-up target games are 2-D shoot across the screen type, and quite frankly there is a glut of inferior ones. A need for a new approach exists, such as fast 3-D HIRES simulations with clever and complex challenges. How about shooting into the screen, into 3-D space, where the target is mathematically many feet behind the screen surface? How about computer intelligent targets that shoot back and use strategy and learn? How about all this and the best attributes of the more popular games? Let's include high score, 10 levels of play, snappy sound effects, colorful explosions and real time graphics. Why not go all the way and have a three dimensional gunsight? A real space battle simulation. . . Nah. . . no one would believe it or could even write it. Right?
WRONG!! WE HAVE IT. . . and it is SUPER STARBASE GUNNER. We are very excited about this product because it is all the things we wish we had and didn't. And you can have it now with this introductory offer.

SUPER STARBASE GUNNER DISK. . . $19.95 48K with APPLESOFT ROM

AVAILABLE FROM YOUR DEALER OR DIRECT FROM *APPLE—JACK,* BOX 51, CHERRY VALLEY, MA 01611
✓314 (INQUIRIES INVITED)

September, October Super Special
Apple II 16K
$950.00
reg. 1195.00

INTEGRAL DATA SYSTEMS

440G: Paper Tiger with Graphics; 2K Buffer	$950	reg. $1095
460: Word Processing Quality	$1099	reg. 1295
460G: IDS 460 w/Graphics	$1199	reg. 1395

Centronics 737 High Quality Dot Matrix	$895	reg. 995.00
Apple Silentype Includes interface and graphic capabilities	$535	reg. 595.00
Apple Parallel Int.	$160	reg. $180
Apple Serial Int.	$175	reg. $195
Centronics Parallel Int.	$185	reg. $225

DOUBLE VISION	$295.00
DISK II with controller	$525.00
without controller	$445.00
MICROMODEM	$325.00
PASCAL	$425.00
LEEDEX MONITOR	$140.00
KG-12C Green Phosphor 12" Screen w/Glare Cover 18 MHz bandwidth	$275.00

16K RAMS for APPLE II TRS-80 $59

VERBATIM DISKS 10 for $27

The Computer Stop
16919 Hawthorne Blvd.
Lawndale, CA 90260
(213) 371-4010 ✓105

MON. - SAT. 10-6

THE BEST IN SOFTWARE FROM COMPLEAT SYSTEMS

NON-PROFIT/SERVICE INDUSTRY ACCOUNTING

For TRS-80 and CPM. In use over 2 yrs. A partner of a big 8 acct firm said ". . . the best accounting program I have seen . . . does in a few pages what is frequently not done in 50 . . ." Unique features include:

- Twice as fast as other systems
- Common sense accounting—no debits or credits
- Budgets, prior year, and year-to-date
- Current status available at all times
- 8 separate Funds or Co's allowed
- over 2000 accounts allowed
- standard 8½ x 11 output
- one year free update service

Min. system 2 disk, 32K TRS-80 or 1 disk, 48K Z-80 CP/M $695/$35 manual. Complete systems also available.

SECURITY FOR TRS-80 AND CPM

The best security system available. Automatically encodes/decodes all data to/from disk. A billion billion (10^{18}) combinations. Typical uses include: proprietary, sales, financial, tax, or confidential client information; and time sharing/multi user systems such as Source, Micronet, etc.

$49.95 Min. systems 1 disk, 16K, TRS-80 or 24K CP/M.

$500 FREE to first to decode our sample message.

COMPLEAT SYSTEMS 9551 Casaba Ave. Chatsworth, CA 91311 213-993-1479
✓94 Master Charge and VISA accepted. CA residents add 6% tax.
TRS-80 and CPM are registered trademarks of Tandy Corp and Digital Research

kb microcomputing book nook

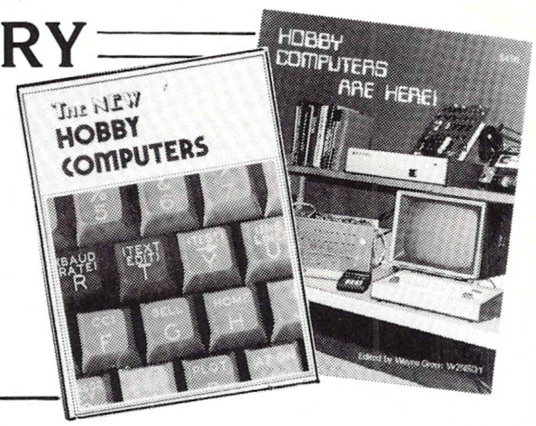
3 NEW BOOKS

● **SOME OF THE BEST FROM KILOBAUD/MICROCOMPUTING**—BK7311—A collection of the best articles that have recently appeared in Kilobaud/MICROCOMPUTING. Included is material on the TRS-80 and PET systems, CP/M, the 8080/8085/Z80 chips, the ASR-33 terminal. Data base management, word processing, text editors and file structures are covered too. Programming techniques and hardcore hardware construction projects for modems, high speed cassette interfaces and TVTs are also included in this large format, 200 plus page edition. $10.95.*

● **UNDERSTANDING AND PROGRAMMING MICROCOMPUTERS**—BK7382—A valuable addition to your computing library. This two part text includes the best articles that have appeared in 73 and Kilobaud Microcomputing magazines on the hardware and software aspects of the new microcomputing hobby. Well known authors and well structured text helps the reader get involved in America's fastest growing hobby. $10.95*

● **40 COMPUTER GAMES**—BK7381—Forty games in all in nine different categories. Games for large and small systems, and even a section on calculator games. Many versions of BASIC used and a wide variety of systems represented. A must for the serious computer gamesman. $7.95*

INTRODUCTORY



● **HOBBY COMPUTERS ARE HERE!**—BK7322—If you (or a friend) want to come up to speed on how computers work . . . hardware and software . . . this is an excellent book. It starts with the fundamentals and explains the circuits, and the basics of programming. This book has the highest recommendations as a teaching aid for newcomers. $4.95.*

● **THE NEW HOBBY COMPUTERS**—BK7340—This book takes it from where "HOBBY COMPUTERS ARE HERE!" leaves off, with chapters on Large Scale Integration, how to choose a microprocessor chip, an introduction to programming, low cost I/O for a computer, computer arithmetic, checking memory boards . . . and much, much more! Don't miss this tremendous value! Only $4.95.*

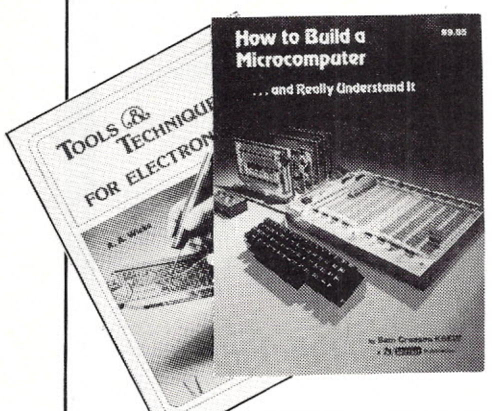
INTRODUCTION TO MICROCOMPUTERS (VOL. 0→III)

● **AN INTRODUCTION TO MICROCOMPUTERS, VOL. 0**—BK1130—The Beginner's Book—Written for readers who know nothing about computers—for those who have an interest in how to use computers—and for everyone else who must live with computers and should know a little about them. The first in a series of 4 volumes, this book will explain how computers work and what they can do. Computers have become an integral part of life and society. During any given day you are affected by computers, so start learning more about them with Volume 0. $7.95.*

● **VOL. I**—BK1030—**2nd Edition completely revised.** Dedicated to the basic concepts of microcomputers and hardware theory. The purpose of Volume I is to give you a thorough understanding of what microcomputers are. From basic concepts (which are covered in detail), Volume I builds the necessary components of a microcomputer system. This book highlights the difference between minicomputers and microcomputers. $12.50.*

● **VOL. II**—BK1040 (with binder)—$30.00* —Contains descriptions of individual microprocessors and support devices used only with the parent microprocessor. Volume II describes all available chips.

● **VOL. III**—BK1133 (with binder)—$20.00.* Contains descriptions of all support devices that can be used with any microprocessor.



● **HOW TO BUILD A MICROCOMPUTER—AND REALLY UNDERSTAND IT**—BK7325—by Sam Creason. The electronics hobbyist who wants to build his own microcomputer system now has a practical "How-To" guidebook. This book is a combination technical manual and programming guide that takes the hobbyist step-by-step through the design, construction, testing and debugging of a complete microcomputer system. Must reading for anyone desiring a true understanding of small computer systems. $9.95.*

● **TOOLS & TECHNIQUES FOR ELECTRONICS**—BK7348—by A. A. Wicks is an easy-to-understand book written for the beginning kit builder as well as the experienced hobbyist. It has numerous pictures and descriptions of the safe and correct ways to use basic and specialized tools for electronic projects as well as specialized metal working tools and the chemical aids which are used in repair shops. $4.95.*

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to Kilobaud *Microcomputing* Book Department • Peterborough NH 03458. Be sure to include check or detailed credit card information. No C.O.D. orders accepted. All orders add $1.00 handling. Please allow 4-6 weeks for delivery. Questions regarding your order? Please write to Customer Service at the above address.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

kb microcomputing book nook

SPECIAL INTERESTS

●**TRS-80 DISK AND OTHER MYSTERIES**—BK1181—by Harvard C. Pennington. This is the definitive work on the TRS-80 disk system. It is full of detailed "How to" information with examples, samples and in-depth explanations suitable for beginners and professionals alike. The recovery of one lost file is worth the price alone. $22.50.*

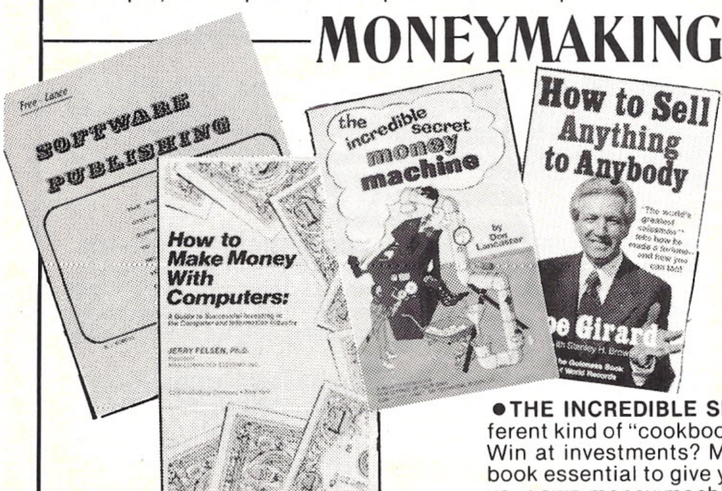
●**INTRODUCTION TO TRS-80 GRAPHICS**—BK1180—by Don Inman. Dissatisfied with your Level I or Level II manual's coverage of graphics capabilities? This well-structured book (suitable for classroom use) is ideal for those who want to use all the graphics capabilities built into the TRS-80. A tutorial method is used with many demonstrations. It is based on the Level I, but all material is suitable for Level II use. $8.95.*

●**MICROPROCESSOR INTERFACING TECHNIQUES**—BK1037—by Austin Lesea & Rodnay Zaks will teach you how to interconnect a complete system and interface it to all the usual peripherals. It covers hardware and software skills and techniques, including the use and design of model buses such as the IEEE 488 or S100. $15.95.*

●**MICROPROCESSOR LEXICON—ACRONYMS AND DEFINITIONS**—BK1137—Compiled by the staff of SYBEX is a convenient reference in pocket size format. Sections include acronyms and definitions, part numbers and their definitions, S-100 signals, RS232 signals, IEEE 499 signals, microcomputer and microprocessors, JETDS summary (military) and a code conversion table. $2.95.*

●**MICROPROCESSORS FROM CHIPS TO SYSTEMS**—BK1036—by Rodnay Zaks is a complete and detailed introduction to microprocessors and microcomputer systems. No preliminary knowledge of computers or microprocessors is required to read this book, although a basic engineering knowledge is naturally an advantage. Intended for all wishing to understand the concepts, techniques and components of microprocessors in a short time. $10.95.*

MONEYMAKING



●**HOW TO MAKE MONEY WITH COMPUTERS**—BK1003—In 10 information-packed chapters, Jerry Felsen describes more than 30 computer-related, money-making, high profit, low capital investment opportunities. $15.00.*

●**HOW TO SELL ANYTHING TO ANYBODY**—BK7306—According to *The Guinness Book of World Records*, the author, Joe Girard, is "the world's greatest salesman." This book reveals how he made a fortune—and how you can, too. $2.25.*

●**FREELANCE SOFTWARE PUBLISHING**—BK1179—by B. J. Korites. "This book is about money and how to make it by writing and selling computer programs," (author's foreword). If you have the skills to write a saleable program, you now need to acquire the skills to sell that program. This compact book comprehensively covers the entire publishing process and many aspects of software salesmanship. $14.95.*

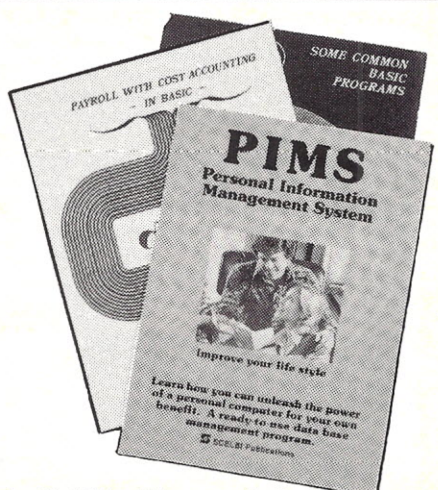
●**THE INCREDIBLE SECRET MONEY MACHINE**—BK1178—by Don Lancaster. A different kind of "cookbook" from Don Lancaster. Want to slash taxes? Get free vacations? Win at investments? Make money from something that you *like* to do? You'll find this book essential to give you the key insider details of what is really involved in starting up your own money machine. $5.95.*

BUSINESS

●**PAYROLL WITH COST ACCOUNTING—IN BASIC**—BK1001—by L. Poole & M. Borchers, includes program listings with remarks, descriptions, discussions of the principle behind each program, file layouts, and a complete user's manual with step-by-step instructions, flowcharts, and simple reports and CRT displays. Payroll and cost accounting features include separate payrolls for up to 10 companies, time-tested interactive data entry, easy correction of data entry errors, job costing (labor of distribution), check printing with full deduction and pay detail, and 16 different printed reports, including W-2 and 941 (in **CBASIC**). $20.00.*

●**SOME COMMON BASIC PROGRAMS**—BK1053—published by Adam Osborne & Associates, Inc. Perfect for non-technical computerists requiring ready-to-use programs. Business programs, plus miscellaneous programs. Invaluable for the user who is not an experienced programmer. All will operate in the stand-alone mode. $12.50 paperback.*

●**PIMS: PERSONAL INFORMATION MANAGEMENT SYSTEM**—BK1009—Learn how to unleash the power of a personal computer for your own benefit in this ready-to-use data-base management program. $11.95.*

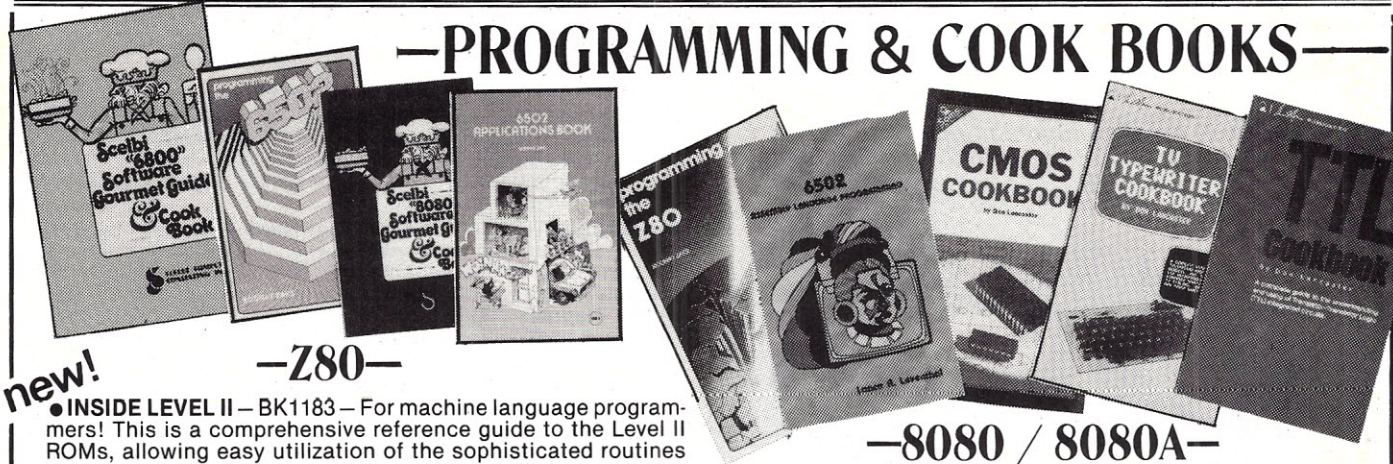


*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to Kilobaud *Microcomputing* Book Department • Peterborough NH 03458. Be sure to include check or detailed credit card information. No C.O.D. orders accepted. All orders add $1.00 handling. Please allow 4–6 weeks for delivery. Questions regarding your order? Please write to Customer Service at the above address.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

kb microcomputing book nook

—PROGRAMMING & COOK BOOKS—



—Z80—

new!

● **INSIDE LEVEL II** — BK1183 — For machine language programmers! This is a comprehensive reference guide to the Level II ROMs, allowing easy utilization of the sophisticated routines they contain. It concisely explains set-ups, calling sequences, variable passage and I/O routines. Part II presents an entirely new composite program structure which unloads under the SYSTEM command and executes in both BASIC and machine code with the speed and efficiency of a compiler. Special consideration is given to disk systems. $15.95.*

● **PROGRAMMING THE Z-80** — BK1122 — by Rodnay Zaks. Here is assembly language programming for the Z-80 presented as a progressive, step-by-step course. This book is both an educational text *and* a self-contained reference book, useful to both the beginning and the experienced programmer who wish to learn about the Z-80. Exercises to test the reader are included. $14.95.*

● **Z-80 ASSEMBLY LANGUAGE PROGRAMMING** — BK1177 — by Lance A. Leventhal. This book thoroughly covers the Z80 instruction set, abounding in simple programming examples which illustrate software development concepts and actual assembly language usage. Features include Z80 I/O devices and interfacing methods, assembler conventions, and comparisons with 8080A/8085 instruction sets and interrupt structure. $12.50.*

● **Z-80 SOFTWARE GOURMET GUIDE AND COOKBOOK** — BK1045 — by Nat Wadsworth. Scelbi's newest cookbook! This book contains a complete description of the powerful Z-80 instruction set and a wide variety of programming information. Use the author's ingredients including routines, subroutines and short programs, choose a time-tested recipe and start cooking! $15.95.*

—6502—

● **PROGRAMMING THE 6502 (Second Edition)** — BK1005 — Rodnay Zaks has designed a self-contained text to learn programming, using the 6502. It can be used by a person who has never programmed before, and should be of value to anyone using the 6502. The many exercises will allow you to test yourself and practice the concepts presented. $12.95.*

● **6502 APPLICATIONS BOOK** — BK1006 — Rodnay Zaks presents practical-application techniques for the 6502 microprocessor, assuming an elementary knowledge of microprocessor programming. You will build and design your own domestic-use systems and peripherals. Self-test exercises included. $12.95.*

● **6502 ASSEMBLY LANGUAGE PROGRAMMING** — BK1176 — by Lance A. Leventhal. This book provides comprehensive coverage of the 6502 microprocessor assembly language. Leventhal covers over 80 programming examples from simple memory load loops to complete design projects. Features include 6502 assembler conventions, input/output devices and interfacing methods, and programming the 6502 interrupt system. $12.50.*

● **6502 SOFTWARE GOURMET GUIDE AND COOKBOOK** — BK1055 — by Robert Findley. This book introduces the BASIC language programmer into the realm of machine-language programming. The description of the 6502 structure and instruction set, various routines, subroutines and programs are the ingredients in this cookbook. "Recipes" are included to help you put together exactly the programs to suit your taste. $12.95.*

—8080 / 8080A—

● **8080A/8085 Assembly Language Programming**—by Lance Leventhal—BK1004—Assembly language programming for the 8080A/8085 is explained with a description of the functions of assemblers and assembly instructions, and a discussion of basic software development concepts. Many fully debugged, practical programs are included as is a special section on structured programming. $12.50.*

● **8080 PROGRAMMING FOR LOGIC DESIGN**—BK1078—Ideal reference for an indepth understanding of the 8080 processor. Application-oriented and the 8080 is discussed in light of replacing conventional, hard-wired logic. Practical design considerations are provided for the implementation of an 8080-base control system. $9.50.*

● **8080 SOFTWARE GOURMET GUIDE AND COOKBOOK**—BK1102—If yu have been spending too much time developing simple routines for your 8080, try this new book by Scelbi Computing and Robert Findley. Describes sorting, searching, and many other routines for the 8080 user. $12.95.*

—6800—

● **6800 PROGRAMMING FOR LOGIC DESIGN**—BK1077—Oriented toward the industrial user, this book describes the process by which conventional logic can be replaced by a 6800 microprocessor. Provides practical information that allows an experimenter to design a complete micro control system from the "ground up." $9.50.*

● **6800 SOFTWARE GOURMET GUIDE AND COOKBOOK** — BK1075 — Like its culinary cousin, *The 8080 Gourmet Guide*, this book by Scelbi Computing and Robert Findley describes sorting, searching and other routines — this time for the 6800 user. $12.95.*

—COOK BOOKS—

● **CMOS COOKBOOK** — BK1011 — by Don Lancaster. Details the application of CMOS, the low power logic family suitable for most applications presently dominated by TTL. Required reading for every serious digital experimenter! $10.50.*

● **TVT COOKBOOK** — BK1064 — by Don Lancaster. Describes the use of a standard television receiver as a microprocessor CRT terminal. Explains and describes character generation, cursor control and interface information in typical, easy-to-understand Lancaster style. $9.95.*

● **TTL COOKBOOK** — BK1063 — by Donald Lancaster. Explains what TTL is, how it works, and how to use it. Discusses practical applications, such as a digital counter and display system, events counter, electronic stopwatch, digital voltmeter and a digital tachometer. $9.50.*

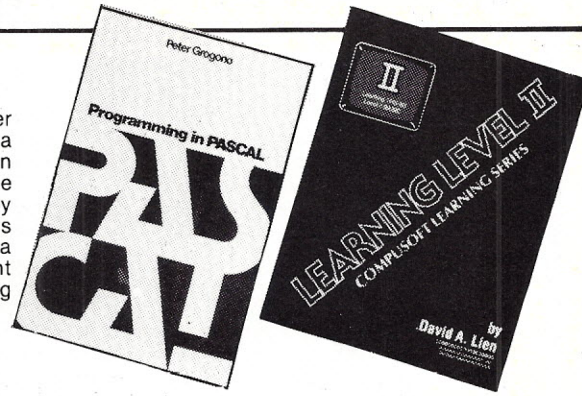
● ***MICROCOMPUTING* CODING SHEETS** ***Microcomputing***'s dozen or so programmers wouldn't try to work without these handy scratch pads, which help prevent the little errors that can cost hours and hours of programming time. Available for programming is Assembly/Machine Language (PD1001), which has columns for address, instruction (3 bytes), source code (label, op code, operand) and comments; and for BASIC (PD1002) which is 72 columns wide. 50 sheets to a pad. $2.39.*

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to Kilobaud *Microcomputing* Book Department • Peterborough NH 03458. Be sure to include check or detailed credit card information. No C.O.D. orders accepted. All orders add $1.00 handling. Please allow 4-6 weeks for delivery. Questions regarding your order? Please write to Customer Service at the above address.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

kb microcomputing book nook

BASIC AND PASCAL



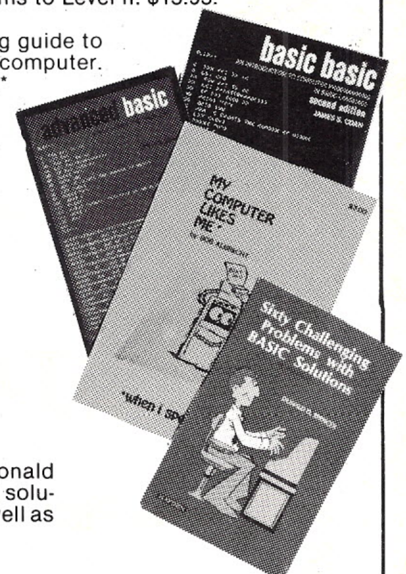
NEW REVISED EDITION

● **PROGRAMMING IN PASCAL**—BK1140—by Peter Grogono. The computer programming language PASCAL was the first language to embody in a coherent way the concepts of structured programming, which has been defined by Edsger Dijkstra and C.A.R. Hoare. As such, it is a landmark in the development of programming languages. PASCAL was developed by Niklaus Wirth in Zurich; it is derived from the language ALGOL 60 but is more powerful and easier to use. PASCAL is now widely accepted as a useful language that can be efficiently implemented, and as an excellent teaching tool. It does not assume knowledge of any other programming language; it is therefore suitable for an introductory course. $12.95.*

● **THE BASIC HANDBOOK**—BK1174—by David Lien. This book is unique. It is a virtual ENCYCLOPEDIA of BASIC. While not favoring one computer over another, it explains over 250 BASIC words, how to use them and alternate strategies. If a computer does not possess the capabilities of a needed or specified word, there are often ways to accomplish the same function by using another word or combination of words. That's where the HANDBOOK comes in. It helps you get the most from your computer, be it a "bottom-of-the-line" micro or an oversized monster. $14.95.*

● **LEARNING LEVEL II**—BK1175—by David Lien. Written especially for the TRS-80, this book concentrates on Level II BASIC, exploring every important BASIC language capability. Updates are included for those who have studied the Level I User's Manual. Sections include: how to use the Editor, dual cassette operation, printers and peripheral devices, and the conversion of Level I programs to Level II. $15.95.*

● **BASIC NEW 2ND EDITION**—BK1081—by Bob Albrecht. Self-teaching guide to the computer language you will need to know for use with your microcomputer. This is one of the easiest ways to learn computer programming. $6.95.*



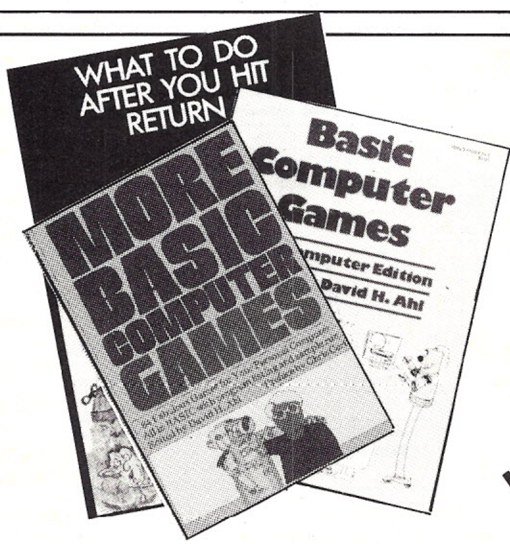
● **BASIC BASIC (2ND EDITION)**—BK1026—by James S. Coan. This is a textbook which incorporates the learning of computer programming using the BASIC language with the teaching of mathematics. Over 100 sample programs illustrate the techniques of the BASIC language and every section is followed by practical problems. This second edition covers character string handling and the use of data files. $9.45.*

● **ADVANCED BASIC**—BK1000—Applications, including strings and files, coordinate geometry, area, sequences and series, simulation, graphing and games. $9.65*.

● **MY COMPUTER LIKES ME... WHEN I SPEAK BASIC**—BK1039—An introduction to BASIC ... simple enough for kids. If you want to teach BASIC to anyone quickly, this is the way to go. $3.95.*

● **SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS (2nd Edition)**—BK1073—by Donald Spencer, provides the serious student of BASIC programming with interesting problems and solutions. No knowledge of math above algebra required. Includes a number of game programs, as well as programs for financial interest, conversions and numeric manipulations. $6.95.*

GAMES



● **WHAT TO DO AFTER YOU HIT RETURN**—BK1071—PCC's first book of computer games ... 48 different computer games you can play in BASIC ... programs, descriptions, many illustrations. Lunar Landing, Hammurabi, King, Civel 2, Qubic 5, Taxman, Star Trek, Crash, Market, etc. $10.95.*

● **BASIC COMPUTER GAMES**—BK1074—Okay, so once you get your computer and are running in BASIC, then what? Then you need some programs in BASIC, that's what. This book has 101 games for you from very simple to real buggers. You get the games, a description of the games, the listing to put in your computer and a sample run to show you how they work. Fun. Any one game will be worth more than the price of the book for the fun you and your family will have with it. $7.50.*

new!

● **MORE BASIC COMPUTER GAMES**—BK1182—edited by David H. Ahl. More fun in BASIC! 84 new games from the people who brought you *BASIC Computer Games*. Includes such favorites as Minotaur (battle the mythical beast) and Eliza (unload your troubles on the doctor at bargain rates). Complete with game description, listing and sample run. $7.50.*

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to Kilobaud *Microcomputing* Book Department • Peterborough NH 03458. Be sure to include check or detailed credit card information. No C.O.D. orders accepted. All orders add $1.00 handling. Please allow 4-6 weeks for delivery. Questions regarding your order? Please write to Customer Service at the above address.

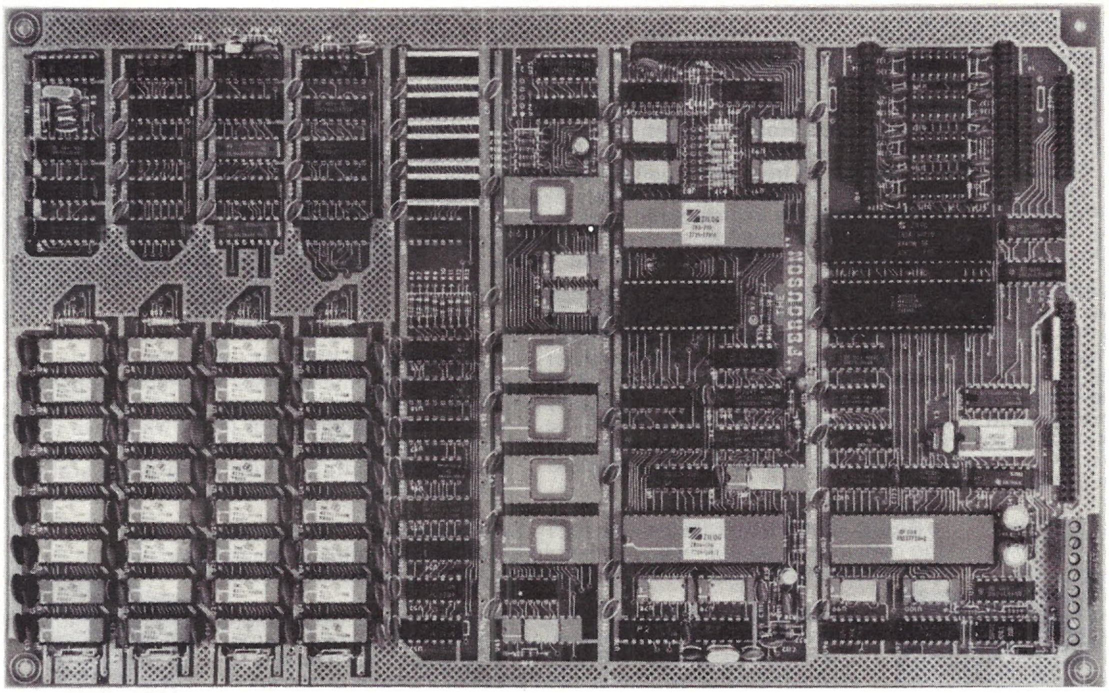
FOR TOLL FREE ORDERING CALL 1-800-258-5473

NEW! "THE BIG BOARD" NEW!

OEM - INDUSTRIAL - BUSINESS - SCIENTIFIC

SINGLE BOARD COMPUTER KIT!

Z-80 CPU! 64K RAM!



THE FERGUSON PROJECT: Three years in the works, and maybe too good to be true. A tribute to hard headed, no compromise, high performance, American engineering! The Big Board gives you all the most needed computing features on one board at a very reasonable cost. The Big Board was designed from scratch to run the latest version of CP/M*. Just imagine all the off-the-shelf software that can be run on the Big Board without any modifications needed! Take a Big Board, add a couple of 8 inch disc drives, power supply, an enclosure, C.R.T., and you have a total Business System for about 1/3 the cost you might expect to pay.

$649.00 ** (64K KIT BASIC I/O)

SIZE: 8½ x 13¾ IN.
SAME AS AN 8 IN. DRIVE.
REQUIRES: +5V @ 3 AMPS
+ - 12V @ .5 AMPS.

FULLY SOCKETED!

FEATURES: (Remember, all this on one board!)

64K RAM

Uses industry standard 4116 RAM'S. All 64K is available to the user, our VIDEO and EPROM sections do not make holes in system RAM. Also, very special care was taken in the RAM array PC layout to eliminate potential noise and glitches.

Z-80 CPU

Running at 2.5 MHZ. Handles all 4116 RAM refresh and supports Mode 2 INTERUPTS. Fully buffered and runs 8080 software.

SERIAL I/O (OPTIONAL)

Full 2 channels using the Z80 SIO and the SMC 8116 Baud Rate Generator. FULL RS232! For synchronous or asynchronous communication. In synchronous mode, the clocks can be transmitted or received by a modem. Both channels can be set up for either data-communication or data-terminals. Supports mode 2 Int. Price for all parts and connectors: $85.

BASIC I/O

Consists of a separate parallel port (Z80 PIO) for use with an ASCII encoded keyboard for input. Output would be on the 80 x 24 Video Display.

24 x 80 CHARACTER VIDEO

With a crisp, flicker-free display that looks extremely sharp even on small monitors. Hardware scroll and full cursor control. Composite video or split video and sync. Character set is supplied on a 2716 style ROM, making customized fonts easy. Sync pulses can be any desired length or polarity. Video may be inverted or true. 5 x 7 Matrix - Upper & Lower Case

FLOPPY DISC CONTROLLER

Uses WD1771 controller chip with a TTL Data Separator for enhanced reliability. IBM 3740 compatible. Supports up to four 8 inch disc drives. Directly compatible with standard Shugart drives such as the SA800 or SA801. Drives can be configured for remote AC off-on. Runs CP/M* 2.2.

TWO PORT PARALLEL I/O (OPTIONAL)

Uses Z-80 PIO. Full 16 bits, fully buffered, bi-directional. User selectable hand shake polarity. Set of all parts and connectors for parallel I/O: $29.95

REAL TIME CLOCK (OPTIONAL)

Uses Z-80 CTC. Can be configured as a Counter on Real Time Clock. Set of all parts: $14.95

SYSTEM COMPARISON

64K RAM KIT	$370.00
80 x 24 Video Kit	365.00
Floppy Disk Controller Kit	235.00
Z-80 CPU Kit	185.95
SER & PAR. I/O	129.95
S-100 Mother Board	45.00
SUB TOTAL	$1330.90

Talk about bangs per buck! The prices shown for S100 kits were taken from the July 1980 BYTE. This will give some basis for comparison between the Big Board and a similar system implementation on the S100 Buss.

CP/M* 2.2 FOR BIG BOARD

The popular CP/M* D.O.S. modified by MICRONIX SYSTEMS to run on Big Board is available for $150.00.

PC BOARD

Blank PC Board with Rom Set and Full Documentation. $195.00

PFM 3.0 2K SYSTEM MONITOR

The real power of the Big Board lies in its PFM 3.0 on board monitor. PFM commands include: Dump Memory, Boot CP/M*, Copy, Examine, Fill Memory, Test Memory, Go To, Read and Write I/O Ports, Disc Read (Drive, Track, Sector), and Search. PFM occupies one of the four 2716 EPROM locations provided.

Z-80 is a Trademark of Zilog.

Digital Research Computers
(OF TEXAS)

P.O. BOX 401565 • GARLAND, TEXAS 75040 • (214) 271-3538

TERMS: Initial shipments will be made approximately 3 to 5 weeks after we receive your order. VISA, MC, cash accepted. We will accept COD's (for the Big Board only) with a $75 deposit. Balance UPS COD. The $75 deposit assures your place in line for the initial production run of Big Board.

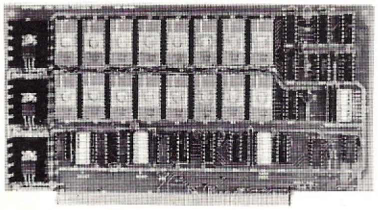
U.K. AND EUROPE: CONTACT MACLIN-ZAND LTD. 38 MOUNT PLEASANT — LONDON WCIX OAP

*TRADEMARK OF DIGITAL RESEARCH. NOT ASSOCIATED WITH DIGITAL RESEARCH OF CALIFORNIA, THE ORIGINATORS OF CPM SOFTWARE

**1 TO 4 PIECE DOMESTIC USA PRICE.

DIGITAL RESEARCH COMPUTERS
(214) 271-3538

32K S-100 EPROM CARD
NEW!



$74.95
KIT

USES 2716's

Blank PC Board - $34

ASSEMBLED & TESTED ADD $30

SPECIAL: 2716 EPROM's (450 NS) Are $14.95 EA. With Above Kit.

KIT FEATURES:
1. Uses +5V only 2716 (2Kx8) EPROM's.
2. Allows up to 32K of software on line!
3. IEEE S-100 Compatible.
4. Addressable as two independent 16K blocks.
5. Cromemco extended or Northstar bank select.
6. On board wait state circuitry if needed.
7. Any or all EPROM locations can be disabled.
8. Double sided PC board, solder-masked, silk-screened.
9. Gold plated contact fingers.
10. Unselected EPROM's automatically powered down for low power.
11. Fully buffered and bypassed.
12. Easy and quick to assemble.

16K DYNAMIC RAM PARTIALS

LOOK! **INTEL 2108 8K X 1 RAMS** LOOK!
8 FOR $9.95 32 FOR $35
FACTORY PRIME!

Huge special purchase of INTEL Dynamic RAM's. These are 2108-4, 300NS, 8K, Ceramic DIP. The 2108 is the INTEL 2116 (16K) tested for either upper or lower 8K only. These are factory prime. Full Spec. See INTEL 1978 Cat. for details or Memory Design Handbook for application data. Both IMSAI and EXTENSYS did mfg. S-100 RAM boards using these devices. — P.S. These devices will not work in the SD EPANDORAM™. Please specify upper or lower 8K. (S1626 or S1627). A super easy RAM to interface to a Z80, 16 PIN DIP.

FOR 4MHZ PRICE CUT! **LOW POWER - 300NS** **8 FOR $37.50**

2114 RAM SALE!

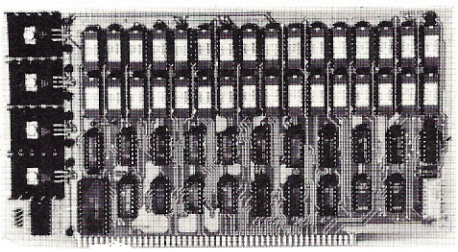
4K STATIC RAM'S. MAJOR BRAND, NEW PARTS.
These are the most sought after 2114's, LOW POWER and 300NS FAST.
8 FOR $37.50

16K STATIC RAM KIT-S 100 BUSS

PRICE CUT!

$199.95 KIT

FOR 4MHZ ADD $10



KIT FEATURES:
1. Addressable as four separate 4K Blocks.
2. ON BOARD BANK SELECT circuitry. (Cromemco Standard!). Allows up to 512K on line!
3. Uses 2114 (450NS) 4K Static Rams.
4. ON BOARD SELECTABLE WAIT STATES.
5. Double sided PC Board, with solder mask and silk screened layout. Gold plated contact fingers.
6. All address and data lines fully buffered.
7. Kit includes ALL parts and sockets.
8. PHANTOM is jumpered to PIN 67.
9. LOW POWER: under 1.5 amps TYPICAL from the +8 Volt Buss.
10. Blank PC Board can be populated as any multiple of 4K.

BLANK PC BOARD W/DATA-**$33**
LOW PROFILE SOCKET SET-**$12**
SUPPORT IC'S & CAPS-**$19.95**
ASSEMBLED & TESTED-**ADD $35**

OUR #1 SELLING RAM BOARD!

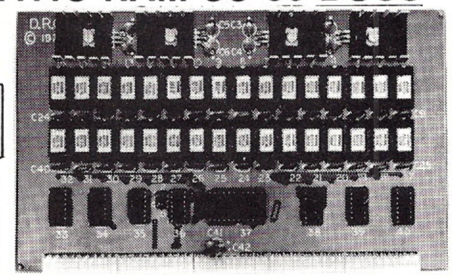
16K STATIC RAM SS-50 BUSS

PRICE CUT!

$210 KIT

FULLY STATIC!

FOR 2MHZ ADD $10



FOR SWTPC 6800 BUSS!

ASSEMBLED AND TESTED - $35

KIT FEATURES:
1. Addressable on 16K Boundaries
2. Uses 2114 Static Ram
3. Fully Bypassed
4. Double sided PC Board. Solder mask and silk screened layout.
5. All Parts and Sockets included
6. Low Power: Under 1.5 Amps Typical

BLANK PC BOARD—**$30** COMPLETE SOCKET SET—**$12**
SUPPORT IC'S AND CAPS—**$19.95**

NEW! STEREO! NEW!
S-100 SOUND COMPUTER BOARD

At last, an S-100 Board that unleashes the full power of two unbelievable General Instruments AY3-8910 NMOS computer sound IC's. Allows you under total computer control to generate an infinite number of special sound effects for games or any other program. Sounds can be called in BASIC, ASSEMBLY LANGUAGE, etc.

COMPLETE KIT! $84.95 (WITH DATA MANUAL)

BLANK PC BOARD W/DATA $31

KIT FEATURES:
- TWO GI SOUND COMPUTER IC'S.
- FOUR PARALLEL I/O PORTS ON BOARD.
- USES ON BOARD AUDIO AMPS OR YOUR STEREO.
- ON BOARD PROTO TYPING AREA.
- ALL SOCKETS, PARTS AND HARDWARE ARE INCLUDED.
- PC BOARD IS SOLDERMASKED, SILK SCREENED, WITH GOLD CONTACTS.
- EASY, QUICK, AND FUN TO BUILD. WITH FULL INSTRUCTIONS.
- USES PROGRAMMED I/O FOR MAXIMUM SYSTEM FLEXIBILITY

Both Basic and Assembly Language Programming examples are included.

SOFTWARE:
SCL™ is now available! Our Sound Command Language makes writing Sound Effects programs a SNAP! SCL™ also includes routines for Register-Examine-Modify, Memory-Examine-Modify, and Play-Memory. SCL™ is available on CP/M* compatible diskette of 2708 or 2716. Diskette - **$24.95** 2708 - **$19.95** 2716 - **$29.95** Diskette includes the source. EPROM'S are ORG at E000H.

4K DYNAMIC RAM BLOWOUT!
SAME AS INTEL 2107B!
4K RAMS AT AN UNBELIEVABLE 50¢ EACH!!!

Prime, new, National Semi., 1979 date coded, full spec. parts. N.S. #MM5280-5N. Same as INTEL 2107B-4, T.I. TMS4060, NEC uPD411, etc. We bought a HUGE QTY. from a West Coast Distributor at truly DISTRESS PRICES! One of the most popular and reliable RAM's ever made. These parts have been used by almost all Major Computer Main Frame Mfg. the world over! Arranged as 4K x 1, 270 NS Access Time, 22 Pin Dip. These units DO NOT use multiplexed addressing, thus making REFRESH and other timing very simple. See INTEL MEMORY DESIGN HANDBOOK for full application notes. The NAT. SEMI. MEMORY DATA BOOK is available at most Radio Shack Stores. Prime units in original factory tubes!

(With Pin Out Data) **#5280-5N 4096 BITS x 1 270 NS ACCESS**
8 FOR $4.95 32 FOR $16
FACTORY CASE (450 PCS) — $180
Sockets Special: 22 Pin Low Profile (With Purchase of 5280's) 8 FOR $1.

COMPUTER PARTS SPECIALS

74LS175 - **.99**	8035 Intel Single Chip CPU - **5.95**
74LS240 - **1.79**	Signetics 2901 4 Bit Slice - **6.95**
74LS241 - **1.79**	AMD 2903 4 Bit Super Slice - **12.50**
74LS244 - **1.79**	AMD 29705 Dual Port RAM - **8.95**
74LS373 - **1.99**	

NEW! G.I. COMPUTER SOUND CHIP

AY3-8910. As featured in July, 1979 BYTE! A fantastically powerful Sound & Music Generator. Perfect for use with any 8 Bit Microprocessor. Contains: 3 Tone Channels, Noise Generator, 3 Channels of Amplitude Control. 16 bit Envelope Period Control, 2-8 Bit Parallel I/O. 3 D to A Converters, plus much more! All in one 40 Pin DIP. Super easy interface to the S-100 or other busses. **$11.95** **PRICE CUT!**
SPECIAL OFFER: ~~$14.95~~ **each** Add $3 for 60 page Data Manual.

Digital Research Computers
(OF TEXAS)
P.O. BOX 401565 • GARLAND, TEXAS 75040 • (214) 271-3538

TERMS: Add $1.50 postage. We pay balance. Orders under $15 add 75¢ handling. No C.O.D. We accept Visa and MasterCharge. Tex. Res. add 5% Tax. Foreign orders (except Canada) add 20% P & H 90 Day Money Back Guarantee on all items: Orders over $50, add 85¢ for insurance.

*TRADEMARK OF DIGITAL RESEARCH. NOT ASSOCIATED WITH DIGITAL RESEARCH OF CALIFORNIA, THE SUPPLIERS OF CPM SOFTWARE.

Start learning and computing for only ***$129.95*** *with a Netronics 8085-based computer kit. Then expand it in low-cost steps to a business/development system with 64k or more RAM, 8" floppy disk drives, hard disks and multi-terminal I/O.*

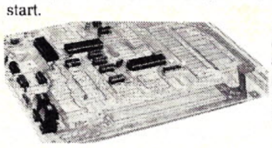
THE NEW EXPLORER/85 SYSTEM

Special! *Full 8" floppy, 64k system for less than the price of a mini! Only* **$1499.95!**

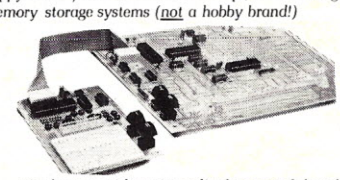
(Also available wired & tested, $1799.95)

Imagine — for only $129.95 you can own the starting level of Explorer/85, a computer that's expandable into full business/development capabilities — a computer that can be your beginner system, an OEM controller, or an IBM-formatted 8" disk small business system. From the first day you own Explorer/85, you begin computing on a significant level, and applying principles discussed in leading computer magazines. Explorer/85 features the advanced Intel 8085 cpu, which is 100% compatible with the older 8080A. It offers onboard S-100 bus expansion, Microsoft BASIC in ROM, plus instant conversion to mass storage disk memory with standard IBM-formatted 8" disks. All for only $129.95, plus the cost of power supply, keyboard/terminal and RF modulator if you don't have them (see our remarkable prices below for these and other accessories). With a Hex Keypad/display front panel, Level "A" can be programmed with no need for a terminal, ideal for a controller, OEM, or a real low-cost start.

Full 8" disk system for less than the price of a mini (shown with Netronics Explorer/85 computer and new terminal). System features floppy drive from Control Data Corp., world's largest maker of memory storage systems (not a hobby brand!)



Level "A" is a complete operating system, perfect for beginners, hobbyists, industrial controller use. $129.95



Level "A" With Hex Keypad/Display.

LEVEL "A" SPECIFICATIONS

Explorer/85's Level "A" system features the advanced Intel 8085 cpu, an 8355 ROM with 2k deluxe monitor/operating system, and an advanced 8155 RAM I/O . . . all on a single motherboard with room for RAM/ROM/PROM/EPROM and S-100 expansion, plus generous prototyping space.

PC Board: Glass epoxy, plated through holes with solder mask. • **I/O:** Provisions for 25-pin (DB25) connector for terminal serial I/O, which can also support a paper tape reader . . . cassette tape recorder input and output . . . cassette tape control output . . . LED output indicator on SOD (serial output) line . . . printer interface (less drivers) . . . total of four 8-bit plus one 6-bit I/O ports. • **Crystal Frequency:** 6.144 MHz. • **Control Switches:** Reset and user (RST 7.5) interrupt . . . additional provisions for RST 5.5, 6.5 and TRAP interrupts onboard. • **Counter/Timer:** Programmable, 14-bit binary. • **System RAM:** 256 bytes located at F800, ideal for smaller systems and for use as an isolated stack area in expanded systems . . . RAM expandable to 64K via S-100 bus or 4k on motherboard.

System Monitor (Terminal Version): 2k bytes of deluxe system monitor ROM located at FØØØ, leaving ØØØØ free for user RAM/ROM. Features include tape load with labeling . . . examine/change contents of memory . . . insert data . . . warm start . . . examine and change all registers . . . single step with register display at each break point, a debugging/training feature . . . go to execution address . . . move blocks of memory from one location to another . . . fill blocks of memory with a constant . . . display blocks of memory . . . automatic baud rate selection to 9600 baud . . . variable display line length control (1-255 characters/line) . . . channelized I/O monitor routine with 8-bit parallel output for high-speed printer . . . serial console in and console out channel so that monitor can communicate with I/O ports.

System Monitor (Hex Keypad/Display Version): Tape load with labeling . . . tape dump with labeling . . . examine/change contents of memory . . . insert data . . . warm start . . . examine and change all registers . . . single step with register display at each break point . . . go to execution address. Level "A" in this version makes a perfect controller for industrial applications, and is programmed using the Netronics Hex Keypad/Display. It is low cost, perfect for beginners.

HEX KEYPAD/DISPLAY SPECIFICATIONS

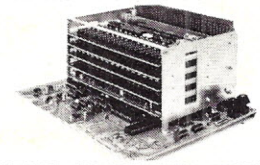
Calculator type keypad with 24 system-defined and 16 user-defined keys. Six digit calculator-type display, that displays full address plus data as well as register and status information.

LEVEL "B" SPECIFICATIONS

Level "B" provides the S-100 signals plus buffers/drivers to support up to six S-100 bus boards, and includes: address decoding for onboard 4k RAM expansion selectable in 4k blocks . . . address decoding for onboard 8k EPROM expansion selectable in 8k blocks . . . address and data bus drivers for onboard expansion . . . wait state generator (jumper selectable), to allow the use of slower memories . . . two separate 5 volt regulators.

LEVEL "C" SPECIFICATIONS

Level "C" expands Explorer/85's motherboard with a card cage, allowing you to plug up to six S-100 cards directly into the motherboard. Both cage and card are neatly contained inside Explorer's deluxe steel cabinet. Level "C" includes a sheet metal superstructure, a 5-card, gold plated S-100 extension PC board that plugs into the motherboard. Just add required number of S-100 connectors.



Explorer/85 With Level "C" Card Cage.

LEVEL "D" SPECIFICATIONS

Level "D" provides 4k of RAM, power supply regulation, filtering decoupling components and sockets to expand your Explorer/85 memory to 4k (plus the original 256 bytes located in the 8155A). The static RAM can be located anywhere from ØØØØ to EFFF in 4k blocks.

LEVEL "E" SPECIFICATIONS

Level "E" adds sockets for 8k of EPROM to use the popular Intel 2716 or the TI 2516. It includes all sockets, power supply regulator, heat sink, filtering and decoupling components. Sockets may also be used for 2k x 8 RAM IC's (allowing for up to 12k of onboard RAM).

DISK DRIVE SPECIFICATIONS

- 8" CONTROL DATA CORP. professional drive.
- LSI controller.
- Write protect.
- Single or double density.
- Data capacity: 401,016 bytes (SD), 802,032 bytes (DD), unformatted.
- Access time: 25ms (one track).

DISK CONTROLLER/ I/O BOARD SPECIFICATIONS

- Controls up to four 8" drives.
- 1771A LSI (SD) floppy disk controller.
- Onboard data separator (IBM compatible).
- 2 Serial I/O ports
- Autoboot to disk system when system reset.
- 2716 PROM socket included for use in custom applications.
- Onboard crystal controlled.
- Onboard I/O baud rate generators to 9600 baud.
- Double-sided PC board (glass epoxy.)

DISK DRIVE CABINET/POWER SUPPLY

- Deluxe steel cabinet with individual power supply for maximum reliability and stability.

ORDER A COORDINATED EXPLORER/85 APPLICATIONS PAK!

Beginner's Pak (Save $26.00!) — Buy Level "A" (Terminal Version) with Monitor Source Listing and AP-1 5-amp Power Supply: (regular price $199.95), now at SPECIAL PRICE: **$169.95** plus post. & insur.

Experimenter's Pak II (Save $53.40!) — Buy Level "A" (Hex Keypad/Display Version) with Hex Keypad/Display, Intel 8085 User Manual, Level "A" Hex Monitor Source Listing, and AP-1 5-amp Power Supply: (regular price $279.35), all at SPECIAL PRICE: **$219.95** plus post. & insur.

Special Microsoft BASIC Pak (Save $103.00!) — Includes Level "A" (Terminal Version), Level "B", Level "D" (4k RAM), Level "E", 8k Microsoft in ROM, Intel 8085 User Manual, Level "A" Monitor Source Listing, and AP-1 5-amp Power Supply: (regular price $439.70), now yours at SPECIAL PRICE: **$329.95** plus post. & insur.

ADD A TERMINAL WITH CABINET, GET A FREE RF MODULATOR: Save over $114 at this SPECIAL PRICE: **$499.95** plus post. & insur.

Special 8" Disk Edition Explorer/85 (Save over $104!) — Includes disk-version Level "A", Level "B", two S-100 connectors and brackets, disk controller, 64k RAM, AP-1 5-amp power supply, Explorer/85 deluxe steel cabinet, cabinet fan, 8" SD/DD disk drive from famous CONTROL DATA CORP. (not a hobby brand!), drive cabinet with power supply, and drive cable set-up for two drives. This package includes everything but terminal and printers (see coupon for them). Regular price $1630.30, all yours in kit at SPECIAL PRICE: **$1499.95** plus post. & insur. Wired and tested, only **$1799.95.**

Special! Complete Business Software Pak (Save $625.00!) — Includes CP/M 2.0, Microsoft BASIC, General Ledger, Accounts Receivable, Accounts Payable, Payroll Package: (regular price $1325), yours now at SPECIAL PRICE: **$699.95.**

Dept. 11K ***Please send the items checked below:***

- ☐ **Explorer/85 Level "A" kit** (Terminal Version) . . . **$129.95** plus $3 post. & insur.
- ☐ **Explorer/85 Level "A" kit** (Hex Keypad/Display Version) . . . **$129.95** plus $3 post. & insur.
- ☐ **8k Microsoft BASIC** on cassette tape. **$64.95** postpaid.
- ☐ **8k Microsoft BASIC in ROM kit** (requires Levels "B", "D" and "E") . . . **$99.95** plus $2 post. & insur.
- ☐ **Level "B" (S-100) kit** . . . **$49.95** plus $2 post. & insur.
- ☐ **Level "C" (S-100 6-card expander) kit** . . . **$39.95** plus $2 post. & insur.
- ☐ **Level "D" (4k RAM) kit** . . . **$69.95** plus $2 post. & insur.
- ☐ **Level "E" (EPROM/ROM) kit** . . . **$5.95** plus 50¢ p&h.
- ☐ **Deluxe Steel Cabinet** for Explorer/85 . . . **$49.95** plus $3 post. & insur.
  - ☐ Fan For Cabinet . . . **$15.00** plus $1.50 post. & insur.
- ☐ **ASCII Keyboard/Computer Terminal kit:** features a full 128 character set, u&l case; full cursor control; 75 ohm video output; convertible to baudot output; selectable baud rate, RS232-C or 20 ma. I/O, 32 or 64 character by 16 line formats, and can be used with either a CRT monitor or a TV set (if you have an RF modulator) . . . **$149.95** plus $3.00 post. & insur.
  - ☐ **DeLuxe Steel Cabinet** for ASCII keyboard/terminal . . . **$19.95** plus $2.50 post. & insur.
- ☐ **New! Terminal/Monitor:** (See photo) Same features as above, except 12" monitor with keyboard and terminal is in deluxe single cabinet; kit . . . **$399.95** plus $7 post. & insur.
- ☐ **Hazeltine terminals:** Our prices too low to quote — **CALL US**
- ☐ **Lear-Sigler terminals/printers:** Our prices too low to quote: **CALL US**
- ☐ **Hex Keypad/Display kit** . . . **$69.95** plus $2 post. & insur.
- ☐ **AP-1 Power Supply Kit** ±8V @ 5 amps) in deluxe steel cabinet . . . **$39.95** plus $2 post. & insur.
- ☐ **Gold Plated S-100 Bus Connectors** . . . **$4.85** each, postpaid.
- ☐ **RF Modulator kit** (allows you to use your TV set as a monitor) . . . $8.95 **postpaid.**
- ☐ **16k RAM kit** (S-100 board expands to 64k) . . . **$199.95** plus $2 post. & insur.
- ☐ **32k RAM kit** . . . **$299.95** plus $2 post. & insur.
- ☐ **48k RAM kit** . . . **$399.95** plus $2 post. & insur.
- ☐ **64k RAM kit** . . . **$499.95** plus $2 post. & insur.
- ☐ **16k RAM Expansion kit** (to expand any of the above in 16k blocks up to 64k) . . . **$99.95** plus $2 post. & insur. each.
- ☐ **Intel 8085 cpu Users' Manual** . . . **$7.50** postpaid.
- ☐ **12" Video Monitor** (10MHz bandwidth) . . . **$139.95** plus $5 post. & insur.
- ☐ **Beginner's Pak** (see above) **$169.95** plus $4 post. & insur.
- ☐ **Experimenter's Pak** (see above) . . . **$219.95** plus $6 post. & insur.
- ☐ **Special Microsoft BASIC Pak Without Terminal** (see above) . . . **$329.95** plus $7 post. & insur.
- ☐ Same as above, plus **ASCII Keyboard Terminal With Cabinet, Get Free RF Modulator** (see above) . . . **$499.95** plus $10 post. & insur.
- ☐ **Special 8" Disk Edition Explorer/85** (see above) . . . **$1499.95** plus $26 post. & insur.
  - ☐ **Wired & Tested** . . . **$1799.95** plus $26 post. & insur.
- ☐ **Extra 8" CDC Floppy Drives** . . . **$499.95** plus $12 post. & insur.
  - ☐ **Cabinet & Power Supply For Drive** . . . **$69.95** plus $3 post. & insur.
- ☐ **Drive Cable Set-up For Two Drives** . . . **$25** plus $1.50 post. & insur.
- ☐ **Disk Controller Board With I/O Ports** . . . **$199.95** plus $2 post. & insur.
- ☐ **Special: Complete Business Software Pak** (see above) . . . **$699.96** postpaid.
  - SOLD SEPARATELY:
  - ☐ **CP/M 1.4** . . . **$100** postpaid.
  - ☐ **CP/M 2.0** . . . **$150** postpaid.
  - ☐ **Microsoft BASIC** . . . **$325** postpaid.
- ☐ **Intel 8085 cpu User Manual** . . . $7.50 **postpaid.**
- ☐ **Level "A" Monitor Source Listing** . . . **$25** postpaid.

Continental U.S.A. Credit Card Buyers Outside Connecticut

CALL TOLL FREE: 800-243-7428

To Order From Connecticut Or For Technical Assistance, call (203) 354-9375

Total Enclosed (Conn res. add sales tax) $ ________

Paid By:

☐ Personal Check ☐ Cashier's Check/Money Order

☐ **VISA** ☐ **Master Charge** (Bank No. _ _ _ _)

Acct. No. ________________ Exp. Date ______

Signature ________________

Print Name ________________

Address ________________

City ________________

State ____________ Zip ____________

NETRONICS Research & Development Ltd.
333 Litchfield Road, New Milford, CT 06776

Get the inside copy of CP/M*

Just purchase a Jade Double-D or CCS double density disk controller this month and CP/M* 2.2 is yours for free.

S-100 Boards

DOUBLE-D - Jade
Double density disk controller with the inside track
IOD-1300K *Kit & CP/M 2.2* **$395.00**
IOD-1300A *8" A & T & CP/M 2.2* .. **$469.00**
IOD-1305A *5¼" A & T & CP/M 2.2* **$469.00**
IOD-1200B *Bare board* **$55.00**

DOUBLE DENSITY - Cal Comp Sys
5¼" or 8" disk controller with free CP/M 2.2
IOD-1400A *A & T* **$374.95**

THE BIG Z* - Jade
2 or 4 MHz switchable Z-80 CPU with serial I/O*
CPU-30201K *Kit* **$145.00**
CPU-30201A *A & T* **$199.00**
CPU-30200B *Bare board* **$35.00**

SBC-100 - SD Systems
2.5 MHz Z-80 CPU with serial & parallel I/O ports*
CPC-30100K *Kit* **$269.95**
CPC-30100A *Jade A & T* **$339.95**

SBC-200 - SD Systems
4 MHz Z-80 CPU with serial & parallel I/O ports*
CPC-30200K *Kit* **$299.95**
CPC-30200A *Jade A & T* **$375.00**

CB2 - S.S.M.
2 or 4 MHz switchable Z-80 CPU with RAM, ROM, & I/O*
CPU-30300K *Kit* **$239.95**
CPC-30300A *A & T* **$299.95**

2810 Z-80* CPU - Cal Comp Sys
2/4 MHz Z-80A CPU w/serial I/O port*
CPU-30400A *A & T* **$275.00**

ExpandoRAM I - SD Systems
2.5 MHz RAM board expandable from 16K to 64K
MEM-16130K *16K kit* **$245.00**
MEM-16130A *16K Jade A & T* **$295.00**
MEM-32131K *32K kit* **$275.00**
MEM-32131A *32K Jade A & T* **$325.00**
MEM-48132K *48K kit* **$305.00**
MEM-48132A *48K Jade A & T* **$355.00**
MEM-64133K *64K kit* **$335.00**
MEM-64133A *64K Jade A & T* **$385.00**

64K RAM BOARD $359.95

ExpandoRAM II - SD Systems
4 MHz RAM board expandable from 16K to 256K
MEM-16630A *16K kit* **$249.95**
MEM-16630A *16K Jade A & T* **$299.95**
MEM-32631K *32K kit* **$289.95**
MEM-32631A *32K Jade A & T* **$339.95**
MEM-48632K *48K kit* **$324.95**
MEM-48631A *48K Jade A & T* **$374.95**
MEM-64633K *64K kit* **$359.95**
MEM-64633A *64K Jade A & T* **$409.95**

32K STATIC RAM BOARD $299.95
16K STATIC RAM BOARD $169.95

32K STATIC RAM - Jade
2 or 4 MHz expandable static RAM board uses 2114L's
MEM-16151K *16K 4 MHz kit* **$169.95**
MEM-16151A *16K 4 MHz A & T* ... **$224.95**
MEM-32151K *32K 4 MHz kit* **$299.95**
MEM-32151A *32K 4 MHz A & T* ... **$349.95**

S.P.I.C. - Jade
Our new I/O card with 2 SIO's, 4 CTC's, and 1 PIO
IOI-1045K *2 CTC's, 1 SIO, 1 PIO* .. **$199.00**
IOI-1045A *A & T* **$259.00**
IOI-1046K *4 CTC's, 2 SIO's, 1 PIO* **$259.00**
IOI-1046A *A & T* **$319.00**
IOI-1045B *Bare board w/ manual* ... **$59.95**
IOI-1045D *Manual only* **$20.00**

16K STATIC RAM - Cal Comp Sys
2 or 4 MHz 16K static RAM - a real memory bargain
MEM-16160K *16K 2 MHz kit* **$249.95**
MEM-16160A *16K 2 MHz A & T* ... **$279.00**
MEM-16162K *16K 4 MHz kit* **$279.95**
MEM-16162A *16K 4 MHz A & T* ... **$309.00**
MEM-16160B *Bare board* **$29.95**

PB-1 - S.S.M.
2708, 2716 EPROM board with built-in programmer
MEM-99510K *Kit* **$159.95**
MEM-99510A *A & T* **$239.95**

PROM-100 - SD Systems
2708, 2716, 2732, 2758, & 2516 EPROM programmer
MEM-99520K *Kit* **$175.00**
MEM-99520A *Jade A & T* **$225.00**

I/O-4 - S.S.M.
2 serial I/O ports plus 2 parallel I/O ports
IOI-1010K *Kit* **$179.95**
IOI-1010A *A & T* **$259.95**
IOI-1010B *Bare board* **$35.00**

BIT STREAMER II - Vector Graphic
3 serial I/O ports plus 2 parallel I/O ports
IOI-1025A *A & T* **$259.00**

100K DAY CLOCK - Mtn Hardware
Crystal controlled S-100 clock with NiCad backup
IOK-1400A *A & T* **$329.95**

SB1 - S.S.M.
15 Hz to 25K Hz music synthesizer for S-100
IOS-1005K *Kit* **$239.95**
IOS-1005A *A & T* **$299.95**

TB-4 - Mullen
Extremely versatile extender board with logic probe
TSX-180K *Kit* **$55.00**
TSX-180A *A & T* **$75.00**

TERMINATOR & EXTENDER - C.C.S.
Can be used as both an S-100 extender and terminator
TSX-150K *Kit* **$39.95**

S-100 EXTENDER - Cal Comp Sys
Puts problem boards within easy reach
TSX-160A *A & T* **$24.95**

VERSAFLOPPY I - SD Systems
Versatile floppy disk controller for 8" or 5¼"
IOD-1150K *Kit* **$219.95**
IOD-1150A *Jade A & T* **$269.95**

VERSAFLOPPY II - SD Systems
New double density controller for both 8" & 5¼"
IOD-1160K *Kit* **$309.95**
IOD-1160A *Jade A & T* **$369.95**

S-100 PROTO BOARD - Jade
Universal design, plated thru holes, gold fingers
TSX-140B *Bare board* **$24.95**

2708/2716 EPROM BOARD - Jade
Holds up to 16 EPROMs, addressable on 1K banks
MEM-16230K *Kit* **$69.95**
MEM-16230A *A & T, no PROMs* **$99.95**
MEM-16230B *Bare board* **$30.00**

***Z-80, Z-80A, and the letter Z are recognized trademarks of Zilog, Inc.**

VDB-8024 - SD Systems
80 x 24 I/O mapped video board with keyboard I/O
IOV-1020K *Kit* **$339.95**
IOV-1020A *Jade A & T* **$399.95**

VB3 - S.S.M.
80 x 24 or 80 x 48 memory mapped with graphics
IOV-1095K *Kit, 4 MHz* **$399.95**
IOV-1095A *A & T, 4 MHz* **$464.95**
IOV-1096K *80 x 48 upgrade, 4 MHz* . **$89.00**

VIDEO BOARD - Jade
64 x 16 assembled & tested S-100 video board
IOV-1050B *Bare board* **$29.95**
IOV-1050A *A & T sale price* **$99.95**

Single Board Computers

AIM-65 - Rockwell
6502 computer with printer, display, & keyboard
CPK-50165 *1K AIM* **$374.95**
CPK-50465 *4K AIM* **$449.95**
SFK-74600008E *8K BASIC ROM* ... **$99.95**
SFK-64600004E *4K assembler ROM* **$84.95**
PSX-030A *Power supply* **$59.95**
ENX-000002 *Enclosure* **$49.95**
4K AIM, 8K BASIC, power supply, & enclosure
Special package price **$599.00**

Z-80* STARTER KIT - SD Systems
Z-80 computer with RAM, ROM, I/O, & keyboard*
CPS-30010K *Kit* **$319.95**
CPS-30010A *Jade A & T* **$399.95**

Video Monitors

VIDEO 100 - Leedex
12" B & W video monitor with 12 MHz bandwidth
VDM-801210 .... **$139.95**

VIDEO 100-80 - Leedex
81 x 24 version of Video 100 with metal cabinet
VDM-801230 .... **$179.95**

9" B & W MONITOR - A.P.F.
High quality, high resolution video monitor
VDM-750900 *9" monitor* **$149.95**

13" COLOR MONITOR - Zenith
The hi res color you've been promising yourself
VDC-201301 .... **$449.00**

12" GREEN SCREEN - NEC
20 MHz, P31 phosphor video monitor with audio
VDM-651200 *12" monitor* **$249.95**

Software

SDOS - SD Systems
DOS, CBASIC-2, Z-80 assembler/editor/linker*
SFX-55001000D *Manual set* **$24.95**
SFX-55001002M *5¼" disks & man* **$199.95**
SFX-55001006F *8" disk & manual* **$199.95**

CP/M 2.2 - Digital Research
Latest & most powerful release of CP/M
SFC-52506000D *Manual set* **$24.95**
SFC-52506000M *5¼" disk & manual* **$149.95**
SFC-52506000F *8" disk & manual* **$149.95**

***CP/M is a registered trademark of Digital Research Corp.**

All royalties paid by Jade Computer Products and California Computer Systems.

track and a free 2.2 to boot !!!

Accessories for Apple

16K MEMORY UPGRADE
Add 16K of RAM to your TRS-80, Apple, or Exidy
MEX-16100K *TRS-80 kit* $39.95
MEX-16101K *Apple kit* $39.95
MEX-16102K *Exidy kit* $39.95

PRINTER INTERFACE - Cal Comp
Centronics type I/O card w/ firmware
IOI-2041A *A & T* $99.95

8" DRIVES for APPLE
Controller, DOS, two 8" drives, cabinet, & cable
Special package price $1475.00

AIO - S.S.M.
Parallel & serial interface for your Apple
IOI-2050K *Kit* $159.00
IOI-2050A *A & T* $199.00

APPLE CLOCK - Cal Comp Sys
Real time clock w/battery back-up
IOK-2100A *A & T* $125.00

SUPERTALKER - Mtn Hardware
Speech recognition/synthesizer w/speaker & mike
IOS-2015A *A & T* $275.00

Z-80* CARD for APPLE
Z-80 CPU card with CP/M for your Apple*
CPX-30800A *A & T* $298.00

MICROMODEM - D.C. Hayes
Auto answer/dial modem card for Apple or S-100
IOM-2010A *Apple modem* $349.95
IOM-1100A *S-100 modem* $375.00

Printers

BASE 2 - Impact Printer
132 cps, bi-directional, tractor feed, & graphics
PRM-13100 .... $675.00

DP-9501 - Anadex
9 x 11 dot matrix, 220 column, 200 cps, & graphics
PRM-10501 *Standard DP-9501* $1495.00
PRM-10511 *with graphics & 2K* .. $1595.00

ANACOM 150 - Special
150 cps, 9 x 9 matrix, tractor feed
PRM-11150 *Parallel interface* $995.00
PRM-11151 *Serial interface* $995.00

SPINWRITER - NEC
65 cps, bi-directional, letter quality with tractor
PRD-55510 *with 2K buffer* $2695.00

NOVATION CAT
300 baud, auto answer/originate acoustic modem
IOM-5200A *Special sale price* $139.00

D-CAT 300 baud, direct connect modem
IOM-5201A *Special sale price* $189.00

EPROM ERASERS
L.S. Engineering UV eraser for up to 48 EPROMs
XME-3200 *A & T* $39.95

Spectronics hi intensity industrial eraser
XME-3100 *Without timer* $69.95
XME-3101 *With timer* $94.50

TV-1 - Best Buy
The inexpensive alternative to video monitors
IOR-5040K *Kit* $8.95

Call for your free 1980 catalog

Disk Drives

JADE DISK PACKAGE
Double-D controller kit, two 8" double density drives CP/M 2.2, cabinet, power supply, & cables
Special package price $1395.00

DUAL 8" DRIVES - Jade
A pair of double density Shugarts in a cabinet
MSF-12800R *2 single sided* $995.00
MSF-125202 *2 double sided* $1425.00

8" DISK DRIVES
Highly reliable double density floppy disk drives
Shugart 801R single sided, double density
MSF-10801R *SA-801R* $425.00
Special Sale Price .... ***2 for $800.00***
Qume Datatrak 8 double sided, double density
MSF-750080 *851R compatible* $625.00

DISKETTES - Jade
Bargain prices on magnificent magnetic media
5¼" single sided, single density, box of 10
MMD-5110103 *Soft sector* $27.95
MMD-5111003 *10 sector* $27.95
MMD-5111603 *16 sector* $27.95
5¼" double sided, double density, box of 10
MMD-5220103 *Soft sector* $39.95
8" single sided, single density, box of 10
MMD-8110103 *Soft sector* $33.95
8" single sided, double density, box of 10
MMD-8120103 *Soft sector* $55.95
8" double sided, double density, box of 10
MMD-8220103 *Soft sector* $57.95

DUAL DISK CABINET Jade
Metal enclosure for 2 801R's, power supply & fan
END-000021 *Bare cabinet* $99.00

Motherboards

ISO-BUS - Jade
Silent, simple, and on sale - a better motherboard
6 Slot (5¼" x 8⅝")
MBS-061B *Bare board* $19.95
MBS-061K *Kit* $39.95
MBS-061A *A & T* $49.95
12 Slot (9¾" x 8⅝")
MBS-121B *Bare board* $29.95
MBS-121K *Kit* $69.95
MBS-121A *A & T* $89.95
18 Slot (14½" x 8⅝")
MBS-181B *Bare board* $49.95
MBS-181K *Kit* $99.95
MBS-181A *A & T* $139.95

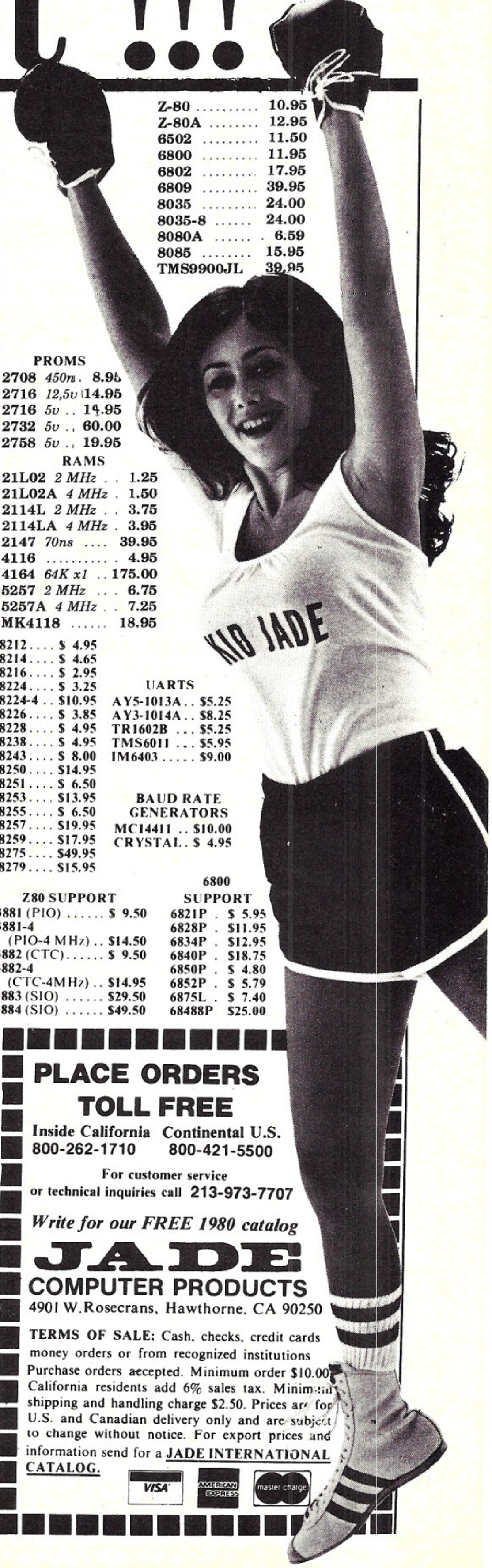
Mainframes

MAINFRAME - Cal Comp Sys
12 slot S-100 mainframe with 20 amp power supply
ENC-112105 *Kit* $309.95
ENC-112106 *A & T* $349.95

DISK MAINFRAME - NNC
Dual 8" drive cutouts with 8 slot motherboard
ENS-112320 *with 30 amp p.s.* $699.95

KIM-1 - Commodore
6502 computer with RAM, ROM, I/O, & keyboard
CPK-50010A *A & T* $175.00

SYM-1 - Synertek
6502 computer with RAM, ROM, I/O, & keyboard
CPK-5002A *A & T* $249.95



Z-80	10.95
Z-80A	12.95
6502	11.50
6800	11.95
6802	17.95
6809	39.95
8035	24.00
8035-8	24.00
8080A	6.59
8085	15.95
TMS9900JL	39.95

PROMS

2708	*450n*	8.95
2716	*12,5v*	14.95
2716	*5v*	14.95
2732	*5v*	60.00
2758	*5v*	19.95

RAMS

21L02	*2 MHz*	1.25
21L02A	*4 MHz*	1.50
2114L	*2 MHz*	3.75
2114LA	*4 MHz*	3.95
2147	*70ns*	39.95
4116		4.95
4164	*64K x1*	175.00
5257	*2 MHz*	6.75
5257A	*4 MHz*	7.25
MK4118		18.95

8212	$ 4.95
8214	$ 4.65
8216	$ 2.95
8224	$ 3.25
8224-4	$10.95
8226	$ 3.85
8228	$ 4.95
8238	$ 4.95
8243	$ 8.00
8250	$14.95
8251	$ 6.50
8253	$13.95
8255	$ 6.50
8257	$19.95
8259	$17.95
8275	$49.95
8279	$15.95

UARTS

AY5-1013A	$5.25
AY3-1014A	$8.25
TR1602B	$5.25
TMS6011	$5.95
IM6403	$9.00

BAUD RATE GENERATORS

MC14411	$10.00
CRYSTAL	$ 4.95

Z80 SUPPORT

3881 (PIO)	$ 9.50
3881-4 (PIO-4 MHz)	$14.50
3882 (CTC)	$ 9.50
3882-4 (CTC-4MHz)	$14.95
3883 (SIO)	$29.50
3884 (SIO)	$49.50

6800 SUPPORT

6821P	$ 5.95
6828P	$11.95
6834P	$12.95
6840P	$18.75
6850P	$ 4.80
6852P	$ 5.79
6875L	$ 7.40
68488P	$25.00

PLACE ORDERS TOLL FREE

Inside California 800-262-1710
Continental U.S. 800-421-5500

For customer service or technical inquiries call **213-973-7707**

Write for our FREE 1980 catalog

JADE COMPUTER PRODUCTS
4901 W.Rosecrans, Hawthorne, CA 90250

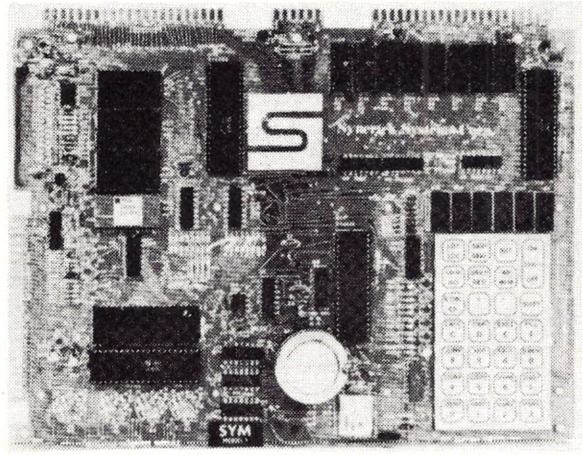
TERMS OF SALE: Cash, checks, credit cards money orders or from recognized institutions Purchase orders accepted. Minimum order $10.00 California residents add 6% sales tax. Minimum shipping and handling charge $2.50. Prices are for U.S. and Canadian delivery only and are subject to change without notice. For export prices and information send for a **JADE INTERNATIONAL CATALOG.**

VISA AMERICAN EXPRESS master charge

SYM-1, 6502-BASED MICROCOMPUTER

- FULLY-ASSEMBLED AND COMPLETELY INTEGRATED SYSTEM that's ready-to-use
- ALL LSI IC'S ARE IN SOCKETS
- 28 DOUBLE-FUNCTION KEYPAD INCLUDING UP TO 24 "SPECIAL" FUNCTIONS
- EASY-TO-VIEW 6-DIGIT HEX LED DISPLAY
- KIM-1* HARDWARE COMPATIBILITY

  The powerful 6502 8-Bit MICROPROCESSOR whose advanced architectural features have made it one of the largest selling "micros" on the market today.
- THREE ON-BOARD PROGRAMMABLE INTERVAL TIMERS available to the user, expandable to five on-board.
- 4K BYTE ROM RESIDENT MONITOR and Operating Programs.
- Single 5 Volt power supply is all that is required.
- 1K BYTES OF 2114 STATIC RAM onboard with sockets provided for immediate expansion to 4K bytes onboard, with total memory expansion to 65, 536 bytes.
- USER PROM/ROM: The system is equipped with 3 PROM/ROM expansion sockets for 2316/2332 ROMs or 2716 EPROMs
- ENHANCED SOFTWARE with simplified user interface
- STANDARD INTERFACES INCLUDE:
  - —Audio Cassette Recorder Interface with Remote Control (Two modes: 135 Baud KIM-1* compatible, Hi-Speed 1500 Baud)
  - —Full duplex 20mA Teletype Interface
  - —System Expansion Bus Interface
  - —TV Controller Board Interface
  - —CRT Compatible Interface (RS-232)
- APPLICATION PORT: 15 Bi-directional TTL Lines for user applications with expansion capability for added lines
- EXPANSION PORT FOR ADD-ON MODULES (51 I/O Lines included in the basic system)
- SEPARATE POWER SUPPLY connector for easy disconnect of the d-c power
- AUDIBLE RESPONSE KEYPAD



Synertek has enhanced KIM-1* software as well as the hardware. The software has simplified the user interface. The basic SYM-1 system is programmed in machine language. Monitor status is easily accessible, and the monitor gives the keypad user the same full functional capability of the TTY user. The SYM-1 has everything the KIM-1* has to offer, plus so much more that we cannot begin to tell you here. So, if you want to know more, the SYM-1 User Manual is available, separately.

SYM-1 Complete w/manuals	**$229.00**
SYM-1 User Manual Only	**$7.00**
SYM-1 Expansion	**$60.00**

Expansion includes 3K of 2114 RAM chips and 1-6522 I/O chip.

SYM-1 Manuals: The well organized documentation package is complete and easy-to-understand.

SYM-1 CAN GROW AS YOU GROW. It's the system to BUILD-ON. Expansion features that are available:

BAS-1 8K Basic ROM (Microsoft Basic)	**$89.00**
KTM-2 (Complete terminal less monitor)	**$319.00**

QUALITY EXPANSION BOARDS DESIGNED SPECIFICALLY FOR KIM-1, SYM-1 & AIM 65

These boards are set up for use with a regulated power supply such as the one below, but, provisions have been made so that you can add onboard regulators for use with an unregulated power supply. But, because of unreliability, we do not recommend the use of onboard regulators. All I.C.'s are socketed for ease of maintenance. *All boards carry full 90-day warranty.*

All products that we manufacture are designed to meet or exceed industrial standards. All components are first qualtiy and meet full manufacturer's specifications. All this and an extended burn-in is done to reduce the normal percentage of field failures by up to 75%. To you, this means the chance of inconvenience and lost time due to a failure is very rare; but, if it should happen, we guarantee a turn-around time of less than forty-eight hours for repair.

Our money back guarantee: If, for any reason you wish to return any board that you have purchased directly from us within ten (10) days after receipt, complete, in original condition, and in original shipping carton; we will give you a complete credit or refund less a $10.00 restocking charge per board.

VAK-1 8-SLOT MOTHERBOARD

This motherboard uses the KIM-4* bus structure. It provides eight (8) expansion board sockets with rigid card cage. Separate jacks for audio cassette, TTY and power supply are provided. Fully buffered bus.

VAK-1 Motherboard **$139.00**

VAK-2/4 16K STATIC RAM BOARD

This board using 2114 RAMs is configured in two (2) separately addressable 8K blocks with individual write-protect switches.

VAK-2 16K RAM Board with only 8K of RAM (½ populated)	**$239.00**
VAK-3 Complete set of chips to expand above board to 16K	**$125.00**
VAK-4 Fully populated 16K RAM	**$325.00**

VAK-5 2708 EPROM PROGRAMMER

This board requires a +5 VDC and ±12 VDC, but has a DC to DC multiplyer so there is no need for an additional power supply. All software is resident in on-board ROM, and has a zero-insertion socket.

VAK-5 2708 EPROM Programmer **$249.00**

VAK-6 EPROM BOARD

This board will hold 8K of 2708 or 2758, or 16K of 2716 or 2516 EPROMs. *EPROMs not included.*

VAK-6 EPROM Board **$119.00**

VAK-7 COMPLETE FLOPPY-DISK SYSTEM **$1299.00**

See May Kilobaud for details

VAK-8 PROTYPING BOARD

This board allows you to create your own interfaces to plug into the motherboard. Etched circuitry is provided for regulators, address and data bus drivers; with a large area for either wire-wrapped or soldered IC circuitry.

VAK-8 Protyping Board **$39.00**

POWER SUPPLIES

ALL POWER SUPPLIES are totally enclosed with grounded enclosures for safety, AC power cord, and carry a full 2-year warranty.

FULL SYSTEM POWER SUPPLY

This power supply will handle a microcomputer and up to 65K of our VAK-4 RAM. ADDITIONAL FEATURES ARE: Over voltage Protection on 5 volts, fused, AC on/off switch. Equivalent to units selling for $225.00 or more.

Provides +5 VDC @ 10 Amps & ±12 VDC @ 1 Amp

VAK-EPS Power Supply **$119.00**

VAK-EPS/AIM—same as VAK-EPS but w/additional 24 volt unregulated (specifically for AIM 65) **$149.00**

KIM-1* Custom P.S. provides 5 VDC @ 1.2 Amps and +12 VDC @ .1 Amps

KCP-1 Power Supply **$39.00**

*KIM is a product of MOS Technology

Add $2.50 for shipping & handling for all <u>except</u> AIM 65.

RNB ENTERPRISES INCORPORATED

✓52

WE'VE EXPANDED AGAIN! NOTE OUR NEW ADDRESS.

2951 W. Fairmount Avenue
Phoenix AZ. 85017
(602)265-7564

VISA · master charge

WAMECO

THE COMPLETE PC BOARD HOUSE
EVERYTHING FOR THE S-100 BUSS

* **FPB-1A** FRONT PANEL BOARD FOR 8080A AND Z80 SYSTEMS IMSAI COMPATIBLE.
  PCBD .... $56.95 KIT $175.00
* **MEM-2** 16K RAM 2114's. ADDRESSABLE IN 4K BOUNDARIES.
  PCBD .... $33.95 KIT (LESS RAMS) $80.95
* **EPM-2** 16/32K ROM USES 2716 OR 2708. ADDRESSABLE IN 4K BOUNDARIES.
  PCBD .... $33.95 KIT (LESS ROMS) $74.95
* **CPU-1** 8080A PROCESSOR BOARD WITH VECTOR INTERRUPT.
  PCBD .... $33.95 KIT $124.95
* **IOB-1** I/O BOARD. ONE SERIAL, TWO PARALLEL WITH CASSETTE. PCBD $33.95
* **FDC-1A** FLOPPY DISC CONTROLLER BOARD USES 1771. PCBD $45.95
* **QMB-12** 13 SLOT MOTHER BOARD.
  PCBD .... $42.95 KIT $125.95
* **QMB-9** 9 SLOT MOTHER BOARD.
  PCBD .... $35.95 KIT $109.95
* **PTB-1** POWER SUPPLY AND TERMINATOR BOARD.
  PCBD .... $29.95 KIT $49.95
* **RTC-1** REAL TIME CLOCK BOARD WITH TWO INTERRUPTS.
  PCBD .... $29.95 KIT $79.95
* **MEM-1A** 8K RAM, USES 2102's.
  PCBD .... $33.95 KIT (LESS RAM) $71.95
* **EPM-1** 4K 170Z BOARD.
  PCBD .... $29.95 KIT (LESS ROM) $59.95

FUTURE PRODUCTS: 80 CHARACTER VIDEO BOARD.
Z-80 CPU BOARD WITH ROM, 8 PARALLEL PORT I/O BOARD.

DEALER INQUIRIES INVITED, UNIVERSITY DISCOUNTS AVAILABLE
AT YOUR LOCAL DEALER

MOST PRODUCTS FOR IMMEDIATE SHIPMENT. NO 4-8 WEEK DELAYS REQUIRED FOR OTHERS.

WMC inc.

WAMECO, INC., P. O. BOX 877 • 455 PLAZA ALHAMBRA • EL GRANADA, CA 94018 • (415) 726-6378

CALIFORNIA COMPUTER SYSTEMS

16K RAM BOARD. Fully buffered addressable in 4K blocks. IEEE standard for bank addressing 2114's. PCBD ....$28.95 Kit 450 NSEC $249.95

PT-1 PROTO BOARD. Over 2,600 holes 4" regulators. All S-100 buss functions labeled, gold fingers. PCBD $28.95

PT-2 PROTO BOARD. Similar to PT-1 except set-up to handle solder tail sockets. PCBD ..$28.95

CCS MAIN FRAME. Kit (S-100) $339.95

APPLE EXTENDER. Kit $22.95

APPLE IEEE INSTRUMENTATION INTERFACE KIT 7490. Kit $275.00

ARITHMETIC PROCESSOR FOR APPLE 7811A. Kit $350.00

APPLE ASYNCHRONOUS SERIAL INTERFACE 7710A. Kit $89.95

APPLE SYNCHRONOUS SERIAL INTERFACE 7712A. Kit $89.95

ALL OTHER CCS PRODUCTS AVAILABLE

SSM

PB-1 2708 & 2716 Programming Board with provisions for 4K or 8K EPROM. No external supplies required. Textool sockets. Kit $143.00

CB-1A 8080 Processor Board. 2K of PROM 256 BYTE RAM power on/rest Vector Jump Parallel port with status. Kit ....$146.00 PCBD $31.95

VB-3 80x24 VIDEO BOARD. Graphics included. 4MHZ $379.95

IO-4 Two serial I/O ports with full handshaking 20/60 ma current loop: Two parallel I/O ports. Kit .... $168.00 PCBD $31.95

VB-IC 64 x 16 video board, upper lower case Greek composite and parallel video with software, S-100. Kit $143.00

CB-2 Z80 CPU BOARD. Kit $199.95

AIO APPLE SERIAL/PARALLEL $144.95

ALL OTHER SSM PRODUCTS AVAILABLE

WMC inc. WAMECO INC.

FDC-1 FLOPPY CONTROLLER BOARD will drive shugart, pertek, remic 5" & 8" drives up to 8 drives, on board PROM with power boot up, will operate with CPM™ (not included). PCBD $43.95

FPB-1 Front Panel. IMSAI size, hex displays. Byte, or instruction single step. PCBD $48.50

MEM-1A 8K x 8 fully buffered, S-100, uses 2102 type rams. PCBD $28.95

QM-12 MOTHER BOARD, 13 slot, terminated, S-100 board only $39.95

CPU-1 8080A Processor board S-100 with 8 level vector interrupt. PCBD $28.95

RTC-1 Realtime clock board. Two independent interrupts. Software programmable. PCBD....... $25.95

EPM-1 1702A 4K Eprom card. PCBD $25.95

EPM-2 2708/2716 16K/32K EPROM CARD. PCBD $28.95

QM-9 MOTHER BOARD. Short Version of QM-12. 9 Slots. PCBD $33.95

MEM-2 16K x 8 Fully Buffered 2114 Board. PCBD $28.95

PTB-1 POWER SUPPLY AND TERMINATOR BOARD. PCBD $28.95

IOB-1 SERIAL AND PARALLEL INTERFACE. 2 parallel, one serial and cassette. PCBD $28.95

2708	$7.50	2114L 450 NSEC	$4.99
2716	$25.95	2114L 200 NSEC	$5.99

MIKOS

(415) 726-7593
P. O. Box 955 • El Granada, CA 94018
Please send for IC, Xistor and Computer parts list

NOV. SPECIAL SALE ON PREPAID ORDERS
(Charge cards not included on this offer)

WAMECO AND MIKOS PARTS SALE.
10% off on Wameco PCBD with Mikos parts assortments.

MIKOS PARTS ASSORTMENT WITH WAMECO AND CYBERCOM PCBDS

MEM-2 with MIKOS #7 16K ram with L2114 450 NSEC $229.95

MEM-2 with MIKOS #13 16K ram with L2114 200 NSEC $249.95

CPU-1 with MIKOS #2 8080A CPU $99.95

QM-12 with MIKOS #4 13 slot mother board $110.95

RTC-1 with MIKOS #5 real time clock........ $65.95

EMP-1 with MIKOS #10 4K 1702 less EPROMS .... $ 49.95

EPM-2 with MIKOS #11 16-32K EPROMS less EPROMS $65.95

QM-9 with MIKOS #12 9 slot mother board $99.95

FPB-1 with MIKOS #14 all parts for front panel $144.95

MIKOS PARTS ASSORTMENTS ARE ALL FACTORY MARKED PARTS. KITS INCLUDE ALL PARTS LISTED AS REQUIRED FOR THE COMPLETE KIT LESS PARTS LISTED. ALL SOCKETS INCLUDED.

LARGE SELECTION OF LS TTL AVAILABLE
PURCHASE $50.00 WORTH OF LS TTL AND GET 10% CREDIT TOWARD ADDITIONAL PURCHASES.
PREPAID ORDERS ONLY.

VISA or MASTERCHARGE. Send account number, interbank number, expiration date and sign your order. Approx. postage will be added. Check or money order will be sent post paid in U.S. If you are not a regular customer, please use charge, cashier's check or postal money order. Otherwise there will be a two-week delay for checks to clear. Calif. residents add 6% tax. Money back 30-day guarantee. We cannot accept returned IC's that have been soldered to. Prices subject to change without notice. **$10 minimum order. $1.50 service charge on orders less than $10.00.**

LIGHT PEN

Comes with Backgammon and Tic-Tac-Toe on tape with full documentation and program listing. Requires 9v. battery. Part No. IBEX $19.95

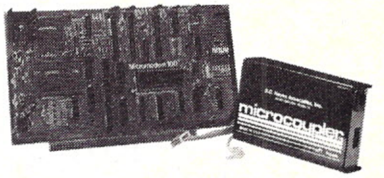
PET COMPUTER

With 16K & monitor—$895.00 • Dual Disk Drive —$1095.00

PARALLEL TRIAC OUTPUT BOARD FOR APPLE II

This board has 8 triacs capable of switching 110 volt 6 amp loads (660 watts per channel) or a total of 5280 watts. Board only $15.00 Part No. 210, with parts $119.95 Part No. 210A

D.C. HAYES MICROMODEM

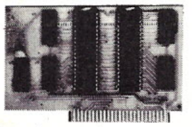


Fully S-100 bus compatible including 16-bit machines and 4 MHz processors. • Two software selectable Baud rates—300 Baud and a jumper selectable speed from 45 to 300 Baud. (110 standard). Supports originate and answer modes. • Direct-connect Microcoupler. This FCC-registered device provides direct access into your local telephone system, with none of the losses or distortions associated with acoustic couplers and without a telephone company supplied data access arrangement. • Auto-Answer/Auto-Call. The MICROMODEM 100 can automatically answer the phone and receive input; it can also dial a number automatically. • Automatic Reset and Disconnect. • Software compatible with the D.C. Hayes Associates 80-103A Data Communications Adapter. Micromodem-DCHA32625—$379.95

APPLE II HOBBY/ PROTOTYPING CARD

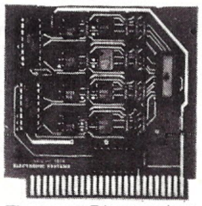
Part No. 7907 $14.95

APPLE II PARALLEL INTERFACE



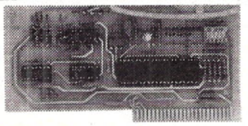
Interfaces printers, synthesizers keyboards, and JBE A-D D-A Converter & Switches. This interface has 4 I/O ports with handshaking logic, 2-6522 VIA's and a 74LS74 for timing. Inputs and outputs are TTL compatible. Part No. 79295K Complete Kit—$69.95 • Part No. 79295A Assembled—$79.95

OPTO-ISOLATED PARALLEL INPUT BOARD FOR APPLE II



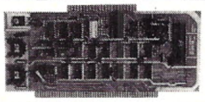
There are 8 inputs that can be driven from TTL logic or any 5 volt source. The circuit board can be plugged into any of the 8 sockets of your Apple II. It has a 16 pin socket for standard dip ribbon cable connection. Board only $15.00. Part No. 120, with parts $69.95. Part No. 120A.

APPLE II✲ SERIAL I/O INTERFACE



Baud rate is continuously adjustable from 0 to 30,000 • Plugs into any peripheral connector • Low current drain. RS-232 input and output • On board switch selectable 5 to 8 data bits, 1 or 2 stop bits, and parity or no parity either odd or even • Jumper selectable address • SOFTWARE • Input and Output routine from monitor or BASIC to teletype or other serial printer • Program for using an Apple II for a video or an intelligent terminal. Also can output in correspondence code to interface with some selectrics. • Also watches DTR • Board only $15.00 Part No. 2, with parts $42.00 Part No. 2A, assembled $62.00 Part No. 2C

TIDMA

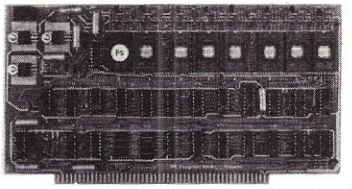


Tape Interface Direct Memory Access • Record and play programs without bootstrap loader (no prom) has FSK encoder/decoder for direct connections to low cost recorder at 1200 baud rate, and direct connections for inputs and outputs to a digital recorder at any baud rate • S-100 bus compatible • Board only $35.00 Part No. 112, with parts $110.00 Part No. 112A.

SYSTEM MONITOR

8080, 8085, or Z-80 System monitor for use with the TIDMA board. There is no need for the front panel. Complete with documentation $12.95.

8K EPROM PICEON

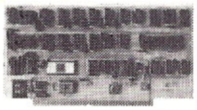


• Programs 2708's address relocation of each 4K of memory to any 4K boundary • Power on jump and reset jump option for "turnkey" systems and computers without a front panel • Program saver software in 1 2708 EPROM $25. Bare board $35 including custom coil, board with parts but no EPROMS $139, with 4 EPROMS $179, with 8 EPROMS $219.

REAL TIME 100,000 DAY CLOCK

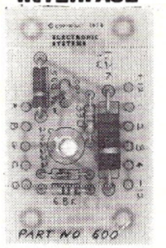
MT. HARDWARE Double the utility of your S-100 bus computer with a real-time clock that keeps time in 100μS increments for over 273 years. Program events for the entire period with real time interrupts...without derailing the system. Maintain a log of computer usage, time and date transaction printouts, call up lists. On-board battery backup. MHPX004—$349.00

VIDEO TERMINAL



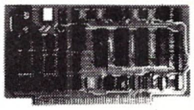
16 lines, 64 columns • Upper and lower case • 5x7 dot matrix • Serial RS-232 in and out with TTL parallel keyboard input • On board baud rate generator 75, 110, 150, 300, 600, & 1200 jumper selectable • Memory 1024 characters (7-21L02) • Video processor chip SFF96364 by Neculonic • Control characters (CR, LF, →, ←, ↑, ↓, non destructive cursor, CS, home, CL • White characters on black background or vice-versa • With the addition of a keyboard, video monitor or TV set with TV interface (part no. 107A) and power supply this is a complete stand alone terminal • also S-100 compatible • requires +16, & −16 VDC at 100mA, and 8VDC at 1A. Part No. 1000A $199.95 kit.

RS-232/TTY INTERFACE



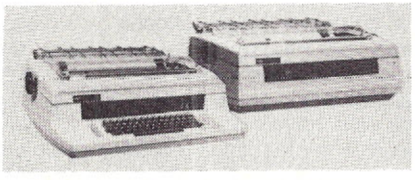
This board has two active circuits, one converts RS-232 to 20 mA, the other converts 20 mA to RS-232. Requires +12 and -12 volts. $9.95 Part No. 600A Kit.

SERIAL I/O



Four Serial I/O RS-232 ports. S-100 Bus, Software or jumper selectable baud rate (110, 300, 600, 1200, 2400, 4800, 9600, 19.2K), on board Xtal baud rate generator, Addressing, switch selectable, Parity or no parity (odd or even) switch selectable, 1 or 2 stop bits, 5 to 8 bits/character. Board only $29.95, Part No. 7908. With parts (kit) $199.95, Part No. 7908A.

SPINWRITER MODELS 5510 and 5520



Features—EIA RS-232C/CCITT V.24 Interface Standard • 55 Characters Per Second Maximum Print Rate • Impeccable Print Quality (OCR Quality) • Microprocessor Electronics • High Resolution Plotting/Graphing • Lowest Operating Noise Level • Self-Test Printing • Operator Engineered Control Panel • Prints Original and up to Seven Copies • NEC Information Systems new Model 5510 Receive Only and Model 5520 Keyboard Send/Receive SPINWRITER terminals are microprocessor controlled serial, impact terminals designed for remote printing applications where impeccable print quality is required. Model 5510 RO, Part No. NECA30759 $2795.95 • Model 5520 KSR, Part No. NECA30762 $3095.95

16K EPROM

Uses 2708 EPROMS, memory speed selection provided, addressable anywhere in 65K of memory, can be shadowed in 4K increments. Board only $24.95 part no. 7902, with parts less EPROMs $49.95 part no. 7902A.

S-100 BUS ACTIVE TERMINATOR

Board only $14.95 Part No. 900, with parts $24.95 Part No. 900A

Send for FREE Catalog...a big self addressed envelope with 80¢ postage gets it fastest!

To Order: Mention part no., description, and price. In USA shipping paid by us for orders accompanied by check or money order. We accept C.O.D. orders (U.S. only) or a VISA or Master Charge no., expiration date, signature and phone no., shipping charges will be added. CA residents add 6.5% for tax. Outside USA add 15% for air mail postage and handling. Payment must be in U.S. dollars. Dealer inquiries invited. Prices subject to change without notice.

Order Line: (408) 448-0800

ELECTRONIC SYSTEMS

Dept. KB, P.O. Box 21638, San Jose, CA USA 95151

Power Supplies! Power Supplies! Power Supplies! SOLID STATE!! (5)

We got 'em! Take your pick . . .

These units are ideal for micro computers. They have been removed from equipment, checked out and guaranteed.

1—5 volts @ 8 amps + 12 volts @ 2 amps + 6 volts @ 75 MA. Power supply has a 3-wire line cord and fused. Dimensions: 10½" × 5½" × 4½". Shipping weight: 16 lbs. . . . . 37.50 ea. 2/70.00

2—Model 818, 5 volts at 15 amps + 12 volts at 4 amps-12 volts at 2 amps. (with line cord). . . . . 35.00 ea. 2/65.00

3— + 5 volts at 5 amps ± 12 volts at 500 ma. + 6 volts at 25 ms. (line cord included). . . . . 32.95 ea. 2/60.00

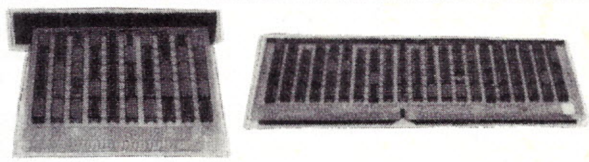
4—Elexon, multi output. Input: 120/240 AC, ±10%, 47-63 hz; output: 1) 12V, 1.5A, DC, OVP; 2) 12V, 1.5A, D.C., OVP. New, in box with operating instructions. 31.50

5—Power Design, Model 1210, constant voltage, DC. P.S. input: 105-125 A.C., 55 to 440 hz. Output: 1-12 volts, 0-10 amps, DC. continuously adjustable output voltage and current limiting. 139.00

COMPUTER GRADE CAPACITORS . . .

18,000 mfd 10 VDC	1.25	11,000 mfd 25 VDC	1.50	4,000 mfd 75 VDC	1.75
4,400 mfd 20 VDC	1.00	35,000 mfd 35 VDC	3.50	1,000 mfd 100 VDC	1.00
46,000 mfd 20 VDC	2.50	10,000 mfd 50 VDC	2.50	6,800 mfd 100 VDC	3.50
3.000 mfd 25 VDC	1.00	22,000 mfd 60 VDC	3.75	4,700 mfd 150 VDC	3.75

WIRE WRAP BOARDS



These boards are pre-wired and removed from equipment. Easy to unwrap for setting up your own board, contains mostly 14-pin IC sockets with individual pin connections. Each board has VCC and ground planes.

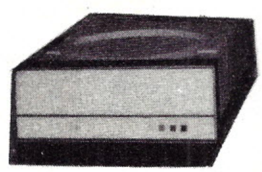
Smaller board measures 6½" × 6" and has 40 to 50 sockets.
Larger board measures 13½" × 6" and has 75 to 100 sockets.

Reduced prices **$7.50 ea. 2/$14.00**
$12.50 ea. 2/$23.00

DIABLO System Disc Drive

SERIES 40, MODEL 43

100 tracks per inch, total capacity of 50 megabits, w/Model 429 power supply, sector counter, 24 sectors, 1 fixed disc, 1 removable disc, average access time 38 ms, PPM: 2600, dimensions: 10 5/16" high, fits in standard rack, equipped with full extension slides, excellent used condition. Shipped freight collect.



$2495

HEWLETT PACKARD model 200CD/rack mounted AUDIO OSCILLATOR freq:5hz to 600khz output: 160mw $165.00

HEWLETT PACKARD model 400D ANALOG VACUUM TUBE VOLTMETER freq: 10hz to 4mhz voltmeter range: 1mv to 300vac in 12 ranges $85.00

TRANSFORMERS

ISOLATION STEP-DOWN TYPE

Primary: 230/115V, 50/60 CPS, Secondary: 115 volts output 250 VA.

$13.95 EACH

IMC MAGNETICS SUPER BOXER FANS

Unused, Model WS2107FL—310, 220/240 VAC, .3 amps, 50/60 hz, 4 11/16" × 4 11/16" × 1 1/2"

$8.95

Clock Crystal Oscillators—TTL, Vectron, type CO-231T. Crystal freq. 4.9152 mhz. Input voltage 5 VDC ±. Output: Drives 10 TTL Loads Logic "0": 0.4V max., sink 16ma. Logic "1" 2.4V min source 2 ma. (above 50 mhz drives 2 Schottky TTL loads). Tuning adjust. with nominal range of ±30 ppm below 25 mhz and 15 ppm above 25 mhz. R.F.E. 1½" × 1½" × ½". $13.95

SG-132 SWEEP SIGNAL GENE OR

FREQ: 15 TO 400 M

Output: AM & FM: CW at any frequency. Crysta mhz or ± 10B. Frequency ac oscilloscope for observing waveforms.

SOLD OUT

$329

TRENDLINE PHONES

Manufactured by I.T.T.

These units have rotary dials. Colors are: white, black, red, and green. They are packaged and have 6-foot cord and installation instructions. Used, but in good operating condition.

34.50 WALL TYPE

Minimum order $25.00. Items offered subject to prior sale. FOB, Brockton, Mass. Money order or check w/order. Shipments and handling add 5%. Shipments by parcel post or UPS. No CODs. Mass. residents add 5% sales tax.

WALLEN ELECTRONICS CO. INC. Tel: (617) 588-6440-6441
✓45 108 SAWTELL AVE., BROCKTON, MA. 02402

ELECTRONIC COMPONENTS
TEST EQUIPMENT
CONNECTORS—WIRE

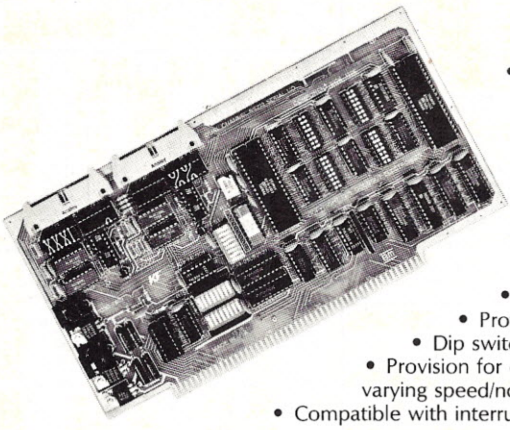
FIRST CLASS Interfacing

CompuPro's feature-packed S-100 I/O boards conform to the IEEE 696/S-100 standard to provide reliable, cost-effective interfacing between your computer and its associated peripherals (such as terminals and printers).

Interfacer I is a dual channel, full RS-232 serial board. Hardward UARTs perform all basic I/O operations, thus freeing the CPU from the need to perform these routines; this increases speed and reliability.

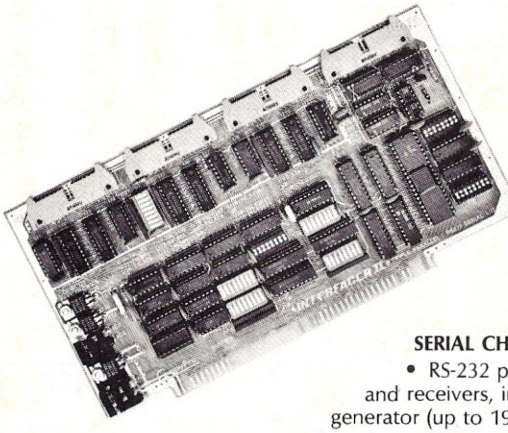
Interfacer II incorporates one channel of serial I/O (identical to an Interfacer I serial port), three full duplex parallel ports for handling I/O data, and a separate full duplex parallel port for status and interrupt control to give unparalleled interfacing flexibility.

Interfacer I



- Dual RS-232 ports with full handshake
- Independently selectable Baud rates for each port, up to 19.2 KBaud – simultaneously drives slow and fast devices (such as teletype/terminal combinations)
- EIA line drivers and receivers
- Conversion to TTL, current loop (20 mA), and RS-232 levels for interfacing to almost any kind of serial device
- On-board crystal timebase for freedom from system clock variations
- Software programmable UART parameters, interrupt enables, and hand shaking lines (handshaking lines are full RS-232 – not just a three wire system)
- Operates with 2 or 4 MHz systems
- Provision for optically isolated current loop for each channel
- Dip switch selectable port addresses
- Provision for custom frequency compensation on both receive and transmit sides (accomodates varying speed/noise situations or unusual cable lengths)
- Compatible with interrupt-driven I/O systems

Interfacer II



PARALLEL CHANNELS

- Latched input and output data with 24 mA drive current
- Each full duplex port has strobe, attention, and enable bits (each with selectable polarity); an input interrupt; and 16 data lines, giving a three port total of 48 true data lines
- Interrupts for each input port
- Separate 25 pin connectors with power for each channel
- Separate status port for interrupt mask and port status
- No mode selection/initialization required
- Handles Centronics type interface, daisy type printers, and interfaces to A/D converters

SERIAL CHANNEL

- RS-232 port includes all features of an Interfacer I serial channel, such as EIA line drivers and receivers, interfacing to almost any kind of serial device, on-board crystal controlled Baud rate generator (up to 19.2 KBaud), full interrupt capability, etc.
- Works with any software I/O drivers developed for the Interfacer I

Either board costs **$199 unkit** (sockets, bypass capacitors pre-soldered in place), **$249 assembled**, and **$324** qualified under the **C**ertified **S**ystem **C**omponent high-reliability program.

These and other CompuPro products are available at finer computer stores world-wide;

write us direct if there's no store in your area.

TERMS: Californians add tax. Allow 5% for shipping, excess refunded. VISA®/Mastercard® orders call **(415) 562-0636**, 24 hours. Please include street address for UPS delivery.

CompuPro™ from **GODBOUT ELECTRONICS** ✓42

Bldg. 725, Oakland Airport, CA 94614

7400

SN7400N	.25			SN74156N	.79
SN7401N	.20			SN74157N	.69
SN7402N	.25			SN74160N	.89
SN7403N	.25	SN7472N	.29	SN74161N	.89
SN7404N	.25	SN7473N	.35	SN74162N	.89
SN7405N	.29	SN7474N	.35	SN74163N	.89
SN7406N	.35	SN7475N	.49	SN74164N	.89
SN7407N	.35	SN7476N	.35	SN74165N	.89
SN7408N	.29	SN7480N	.50	SN74166N	1.25
SN7409N	.29	SN7482N	.99	SN74167N	2.79
SN7410N	.25	SN7483N	.69	SN74170N	1.95
SN7411N	.29	SN7485N	.89	SN74172N	4.95
SN7412N	.35	SN7486N	.35	SN74173N	1.39
SN7413N	.40	SN7489N	1.75	SN74174N	.99
SN7414N	.69	SN7490N	.49	SN74175N	.89
SN7416N	.29	SN7491N	.59	SN74176N	.79
SN7417N	.29	SN7492N	.45	SN74177N	.79
SN7420N	.25	SN7493N	.45	SN74179N	1.49
SN7421N	.29	SN7494N	.69	SN74180N	.79
SN7422N	.45	SN7495N	.69	SN74181N	2.25
SN7423N	.29	SN7496N	.69	SN74182N	.79
SN7425N	.29	SN7497N	3.00	SN74184N	2.49
SN7426N	.29	SN74100N	1.49	SN74185N	2.49
SN7427N	.25	SN74107N	.35	SN74190N	1.25
SN7429N	.39	SN74109N	.39	SN74191N	1.25
SN7430N	.25	SN74116N	1.95	SN74192N	.89
SN7432N	.29	SN74121N	.39	SN74193N	.89
SN7437N	.25	SN74122N	.55	SN74194N	.89
SN7438N	.40	SN74123N	.59	SN74195N	.69
SN7439N	.25	SN74125N	.49	SN74196N	.89
SN7440N	.20	SN74126N	.49	SN74197N	.89
SN7441N	.89	SN74132N	.75	SN74198N	1.49
SN7442N	.59	SN74136N	.75	SN74199N	1.49
SN7443N	1.10	SN74141N	.99	SN74221N	1.25
SN7444N	1.10	SN74142N	3.25	SN74251N	.99
SN7445N	.89	SN74143N	3.49	SN74276N	1.95
SN7446N	.79	SN74144N	3.49	SN74279N	.79
SN7447N	.69	SN74145N	.79	SN74283N	1.49
SN7448N	.79	SN74147N	1.95	SN74284N	3.95
SN7450N	.20	SN74148N	1.29	SN74285N	3.95
SN7451N	.20	SN74150N	1.25	SN74365N	.69
SN7453N	.20	SN74151N	.69	SN74366N	.69
SN7454N	.20	SN74152N	.69	SN74367N	.69
SN7459A	.25	SN74153N	.79	SN74368N	.69
SN7460N	.20	SN74154N	1.25	SN74390N	1.49
SN7470N	.29	SN74155N	.79	SN74393N	1.49

74LS

74LS00	.29			74LS192	1.15
74LS01	.29			74LS193	1.15
74LS02	.29	74LS92	.75	74LS194	1.15
74LS03	.29	74LS93	.75	74LS195	1.15
74LS04	.35	74LS95	.99	74LS197	1.19
74LS05	.35	74LS96	1.15	74LS221	1.19
74LS08	.29	74LS107	.45	74LS240	1.95
74LS09	.35	74LS109	.45	74LS241	1.95
74LS10	.29	74LS112	.45	74LS242	1.95
74LS11	.75	74LS113	.49	74LS243	1.95
74LS12	.35	74LS114	.49	74LS244	1.95
74LS13	.59	74LS122	.89	74LS245	2.95
74LS14	.99	74LS123	1.25	74LS247	1.19
74LS15	.35	74LS125	.89	74LS248	1.19
74LS20	.29	74LS126	.55	74LS249	1.19
74LS21	.35	74LS132	.99	74LS251	1.49
74LS22	.35	74LS133	.89	74LS253	.99
74LS26	.35	74LS136	.49	74LS257	.89
74LS27	.35	74LS138	.89	74LS258	.99
74LS28	.35	74LS139	.89	74LS260	.69
74LS30	.29	74LS151	.89	74LS266	.69
74LS32	.35	74LS153	.89	74LS273	1.95
74LS33	.59	74LS154	1.75	74LS279	.75
74LS37	.45	74LS155	1.19	74LS283	1.09
74LS38	.49	74LS156	1.19	74LS290	.99
74LS40	.35	74LS157	.89	74LS293	.99
74LS42	.89	74LS158	.99	74LS298	1.25
74LS47	.89	74LS160	1.15	74LS352	1.29
74LS48	1.15	74LS161	1.15	74LS353	1.29
74LS49	1.15	74LS162	1.15	74LS365	.75
74LS51	.29	74LS163	1.15	74LS366	.75
74LS54	.29	74LS164	1.25	74LS367	.75
74LS55	.29	74LS165	1.25	74LS368	.75
74LS73	.45	74LS168	1.19	74LS373	1.95
74LS74	.45	74LS169	1.19	74LS374	1.95
74LS75	.59	74LS170	2.49	74LS375	.89
74LS76	.45	74LS173	1.39	74LS386	.69
74LS78	.49	74LS174	.99	74LS393	2.49
74LS83	.89	74LS175	.99	74LS399	2.49
74LS85	1.25	74LS181	2.95	74LS670	2.49
74LS86	.45	74LS190	1.25	81LS95	1.95
74LS90	.59	74LS191	1.25	81LS97	1.95

74S

74S00	.50			74S244	3.25
74S02	.50			74S251	1.45
74S03	.50	74S133	.55	74S253	1.45
74S04	.55	74S134	.69	74S257	1.35
74S05	.55	74S135	1.19	74S258	1.35
74S08	.50	74S136	1.75	74S260	.79
74S09	.50	74S138	1.35	74S280	2.95
74S10	.50	74S139	1.35	74S287*	4.95
74S11	.50	74S140	1.15	74S288*	4.95
74S15	.50	74S151	1.35	74S373	3.49
74S20	.50	74S153	1.35	74S374	3.49
74S22	.50	74S157	1.35	74S387*	5.95
74S30	.50	74S158	1.35	74S472*	19.95
74S32	.55	74S174	1.59	74S473*	19.95
74S40	.55	74S175	1.59	74S474*	21.95
74S51	.50	74S188	4.95	74S475*	21.95
74S64	.50	74S194	1.95	74S570*	7.95
74S65	.50	74S195	1.95	74S571*	7.95
74S74	.79	74S196	3.95	74S572*	19.95
74S86	.79	74S240	2.95	74S573*	19.95
74S112	.79	74S241	2.95	74S574*	14.95
74S113	.79	74S242	3.25	74S940	3.15
74S114	.79	74S243	3.25	74S941	3.15

* LIMITED AVAILABILITY ON THESE PROMS

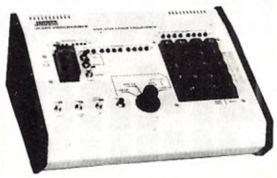
CA—LINEAR

CA3013H	2.15			CA3089N	3.75
CA3023H	3.25			CA3096N	3.95
CA3039H	1.35			CA3130H	1.39
CA3046N	1.30	CA3081N	2.00	CA3140H	1.25
CA3059N	3.25	CA3082N	2.00	CA3160H	1.25
CA3060N	3.25	CA3083N	1.60	CA3401N	.59
CA3080H	1.25	CA3086N	.85	CA3600N	3.50

CD—CMOS

CD4000	.39			CD4082	.39
CD4001	.39			CD4093	.99
CD4002	.39			CD4098	2.49
CD4006	1.19			CD4506	.75
CD4007	.25	CD4041	1.49	CD4507	.99
CD4009	.49	CD4042	.99	CD4508	3.95
CD4010	.49	CD4043	.89	CD4510	1.39
CD4011	.39	CD4044	.89	CD4511	1.29
CD4012	.25	CD4046	1.79	CD4514	3.95
CD4013	.49	CD4047	2.50	CD4515	2.95
CD4014	1.39	CD4048	1.35	CD4516	1.49
CD4015	1.19	CD4049	.49	CD4518	1.79
CD4016	.59	CD4050	.69	CD4519	.89
CD4017	1.19	CD4051	1.19	CD4520	1.29
CD4018	.99	CD4052	1.19	CD4526	1.79
CD4019	.49	CD4053	1.19	CD4528	1.79
CD4020	1.19	CD4056	2.95	CD4529	1.95
CD4021	1.39	CD4059	9.95	CD4543	2.79
CD4022	1.19	CD4060	1.49	CD4562	11.95
CD4023	.29	CD4066	.79	CD4566	2.79
CD4024	.79	CD4068	.39	CD4583	2.49
CD4025	.23	CD4069	.45	CD4723	1.95
CD4026	2.95	CD4070	.55	CD4724	1.95
CD4027	.69	CD4071	.49	MC14409	14.95
CD4028	.89	CD4072	.49	MC14410	14.95
CD4029	1.49	CD4073	.39	MC14411	14.95
CD4030	.49	CD4075	.39	MC14412	11.95
CD4035	.99	CD4076	1.39	MC14419	4.95
CD4040	1.49	CD4081	.39	MC14433	13.95

JE608 PROGRAMMER
2708 EPROM PROGRAMMER



- 3 separate Display Registers: 8 LED's for Hex Key entries, 10 LED's (2^0-2^9) for Address Register and 8 LED's for Data Memory Register. The Data Memory Register displays the content of the RAMs from the EPROM Chip.
- Development of microprocessor systems by means of a ribbon cable from the programmer panel test socket to the EPROM socket on the microprocessor board.
- Rapid checking verification of programmed data changes.
- User may move data from a master to RAM's or write into RAM's with keyboard entries.
- Allows manual stepping manipulation (up and down) at any address location.
- Stand-alone EPROM Programmer consisting of: A 19-key Hexadecimal Keyboard assembly, Programmer Board assembly with 4 power supplies and a LED/Test Socket Panel Board assembly. The Test Socket is zero force insertion type. Power requirements: 115VAC, 60Hz, 6W.
- Compact desk-top enclosure: Color-coordinated designer's case with light tan panels and molded end pieces in mocha brown. Size: 3¼"H x 11"W x 8¾"D. Weight: 5 lbs.

The **JE608 EPROM Programmer** is a completely self-contained unit which is independent of computer control and requires no additional systems for its operations. The EPROM can be programmed from the Hexadecimal Keyboard or from a pre-programmed EPROM. The JE608 Programmer can emulate a programmed EPROM by the use of its internal RAM circuits. This will allow the user to test or pretest a program, for a system, prior to programming a chip. Any changes in the program can be entered directly into the memory circuits with the Hexadecimal Keyboard so that rewriting the entire program will not be necessary. The JE608 Programmer contains a Programmer/Board with 25 IC's and including power supplies of: −5V, +5V, +12V and +26V. The Hexadecimal Keyboard and LED/Test Socket Panel Board are separate assemblies within the system.

JE608K KIT $399.95
JE608A Assembled and tested $499.95

DISCRETE LEDS

XC556R	.200" red	5/$1	MV50	.085" red	6/$1	XC111R	.190" red	5/$1
XC556G	.200" green	4/$1	XC209R	.125" red	5/$1	XC111G	.190" green	4/$1
XC556Y	.200" yellow	4/$1	XC209G	.125" green	4/$1	XC111Y	.190" yellow	4/$1
XC556C	.200" clear	4/$1	XC209Y	.125" yellow	4/$1	XC111C	.190" clear	4/$1
XC22R	.200" red	5/$1	XC526R	.185" red	5/$1			
XC22G	.200" green	4/$1	XC526G	.185" green	4/$1	INFRA-RED LED		
XC22Y	.200" yellow	4/$1	XC526Y	.185" yellow	4/$1	¼"x¼"x1/16" flat		
MV10B	.170" red	4/$1	XC526C	.185" clear	4/$1	IRL — 5/$1		

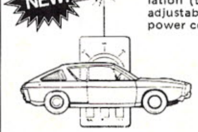
DISPLAY LEDS

C.A. — Common Anode C.C. — Common Cathode

Type	Polarity	Ht	Price	Type	Polarity	Ht	Price
MAN 1	C.A.—red	.270	2.95	DL741	C.A.—red	.600	1.25
MAN 2	5x7 D.M.—red	.300	4.95	DL746	C.A.—red ± 1	.630	1.49
MAN 3	C.C.—red	.125	.25	DL747	C.A.—red	.600	1.49
MAN 52	C.A.—green	.300	1.25	DL750	C.C.—red	.600	1.49
MAN 54	C.C.—green	.300	1.25	DL33B	C.C.—red	.110	.35
MAN 71	C.A.—red	.300	.75	FND70	C.C.	.250	.69
MAN 72	C.A.—red	.300	.75	FND358	C.C. ± 1	.357	.99
MAN 74	C.C.—red	.300	1.25	FND359	C.C.	.357	.75
MAN 81	C.A.—yellow	.300	.49	FND503	C.C. (FND500)	.500	.99
MAN 82	C.A.—yellow	.300	.49	FND507	C.A. (FND510)	.500	.99
MAN 84	C.C.—yellow	.300	.99	HDSP-3401	C.A.—red	.800	1.50
MAN 3620	C.A.—orange	.300	.49	HDSP-3403	C.C.—red	.800	1.50
MAN 3630	C.A.—orange ± 1	.300	.99	5082-7613	C.C., R.H.D.—red	.300	1.25
MAN 3640	C.C.—orange	.300	.99	5082-7620	C.A., L.H.D.—yel.	.300	1.25
MAN 4610	C.A.—orange	.400	.99	5082-7623	C.C., R.H.D.—yel.	.300	1.25
MAN 6610	C.A.—orange—DD	.560	.99	5082-7730	C.A., L.H.D.—red	.300	.99
MAN 6630	C.A.—orange ± 1	.560	.99	5082-7731	C.A., R.H.D.—red	.300	1.25
MAN 6640	C.C.—orange—DD	.560	.99	5082-7750	C.A., L.H.D.—red	.430	1.75
MAN 6650	C.C.—orange ± 1	.560	.99	5082-7751	C.A., R.H.D.—red	.430	1.25
MAN 6660	C.A.—orange	.560	.99	5082-7760	C.C., R.H.D.—red	.430	1.75
MAN 6710	C.A.—red—DD	.560	.99	5082-7300	4x7 sgl. dig. RHD	.600	22.00
MAN 6750	C.C.—red ± 1	.560	.99	5082-7302	4x7 sgl. dig. LHD	.600	22.00
MAN 6780	C.C.—red	.560	.99	5082-7304	Overnge. char. (±1)	.600	19.95
DL704	C.C.—red	.300	1.25	LIT-1	Photo Xsistor Opto-Isol.		.69
DL707	C.A.—red	.300	1.25	MOC3010	Optically Isol. Triac Driver		1.25
DL728	C.C.—red	.500	1.49				

RADIO CONTROL CIRCUITS

NEW!



Ideal for remote control systems which use pulse amplitude modulation (toy cars, boats, tanks, etc.) Features: five function control, adjustable steering angle, suitable for 27 and 47MHz bands and low power consumption.

KB-4428 TRANSMITTER **$4.25**
Abs. max. rating (TA@25°C). Supply volt.: Vcc1 12VDC. Power Dissip.: PD: 300mW; Temp. range: Oper. 0-+50°C — Storage -30— +125° C. Rec. oper. volt.: 7-11V. Crystal or CR Oscillation circuits acceptable.

KB-4429 RECEIVER **$5.95**
Abs. max. rating (TA@25°C). Supply volt.: Vcc1: 11V. Vcc2: 7.5V. Power Dissip.: 600mW. Temp. range: Oper. 0 ± 50°C. Rec. oper. volt.: VOP1 7-11V — VOP2 3-6V.

LOW PROFILE (TIN) SOCKETS

	1-24	25-49	50-100
8 pin LP	.17	.16	.15
14 pin LP	.20	.19	.18
16 pin LP	.22	.21	.20
18 pin LP	.29	.28	.27
20 pin LP	.34	.32	.30
22 pin LP	.37	.36	.35
24 pin LP	.38	.37	.36
28 pin LP	.45	.44	.43
36 pin LP	.60	.59	.58
40 pin LP	.63	.62	.61

SOLDERTAIL STANDARD (TIN)

	1-24	25-49	50-100
14 pin ST	.27	.25	.24
16 pin ST	.30	.27	.25
18 pin ST	.35	.32	.30
24 pin ST	.49	.45	.42
28 pin ST	.99	.90	.81
36 pin ST	1.39	1.26	1.15
40 pin ST	1.59	1.45	1.30

SOLDERTAIL (GOLD) STANDARD

	1-24	25-49	50-100
8 pin SG	.39	.35	.31
14 pin SG	.49	.45	.41
16 pin SG	.54	.49	.44
18 pin SG	.59	.53	.48
24 pin SG	.79	.75	.69
28 pin SG	1.10	1.00	.90
36 pin SG	1.65	1.40	1.26
40 pin SG	1.75	1.59	1.45

WIRE WRAP SOCKETS (GOLD) LEVEL #3

	1-24	25-49	50-100
8 pin WW	.59	.54	.49
10 pin WW	.69	.63	.58
14 pin WW	.79	.73	.67
16 pin WW	.85	.77	.70
18 pin WW	.99	.90	.81
20 pin WW	1.19	1.08	.99
22 pin WW	1.49	1.35	1.23
24 pin WW	1.39	1.26	1.14
28 pin WW	1.69	1.53	1.38
36 pin WW	2.19	1.99	1.79
40 pin WW	2.29	2.09	1.89

1/4 WATT RESISTOR ASSORTMENTS – 5%

ASST. 1	5ea.	10 Ohm 12 Ohm 15 Ohm 18 Ohm 22 Ohm / 27 Ohm 33 Ohm 39 Ohm 47 Ohm 56 Ohm	50pcs.	$1.95
ASST. 2	5ea.	68 Ohm 82 Ohm 100 Ohm 120 Ohm 150 Ohm / 180 Ohm 220 Ohm 270 Ohm 330 Ohm 390 Ohm	50pcs.	$1.95
ASST. 3	5ea.	470 Ohm 560 Ohm 680 Ohm 820 Ohm 1K / 1.2K 1.5K 1.8K 2.2K 2.7K	50pcs.	$1.95
ASST. 4	5ea.	3.3K 3.9K 4.7K 5.6K 6.8K / 8.2K 10K 12K 15K 18K	50pcs.	$1.95
ASST. 5	5ea.	22K 27K 33K 39K 47K / 56K 68K 82K 100K 120K	50pcs.	$1.95
ASST. 6	5ea.	150K 180K 220K 270K 330K / 390K 470K 560K 680K 820K	50pcs.	$1.95
ASST. 7	5ea.	1M 1.2M 1.5M 1.8M 2.2M / 2.7M 3.3M 3.9M 4.7M 5.6M	50pcs.	$1.95
ASST. 8R		Includes Resistor Assts. 1-7 (350 pcs.)		$10.95 ea.

$10.00 Min. Order — U.S. Funds Only
Calif. Residents Add 6% Sales Tax
Postage — Add 5% plus $1 Insurance (if desired)

Spec Sheets — 25¢
1981 Catalog Available — Send 41¢ stamp

NEW 1981 CATALOG

Jameco ELECTRONICS

PHONE ORDERS WELCOME (415) 592-8097

MAIL ORDER ELECTRONICS — WORLDWIDE
1355 SHOREWAY ROAD, BELMONT, CA 94002

11/80 PRICES SUBJECT TO CHANGE

INTERSIL

Part No.	Function	Price
7045IPI	CMOS Precision Timer	14.95
7045EV/Kit*	Stopwatch Chip, XTL	22.95
7106CPL	3½ Digit A/D (LCD Drive)	16.95
7106EV/Kit*	IC, Circuit Board, Display	34.95
7107CPL	3½ Digit A/D (LED Drive)	15.95
7107EV/Kit*	IC, Circuit Board, Display	28.95
7116CPL	3½ Digit A/D LCD Dis. HLD.	18.95
7117CPL	3½ Digit A/D LED Dis. HLD.	17.95
7201IDR	Low Battery Volt Indicator	2.25
7205IPG	CMOS LED Stopwatch/Timer	12.95
7205EV/Kit*	Stopwatch Chip, XTL	19.95
7206CJPE	Tone Generator	5.15
7206CEV/Kit*	Tone Generator Chip, XTL	9.95
7207AIPD	Oscillator Controller	6.50
7207AEV/Kit*	Freq. Counter Chip, XTL	11.10
7208IPI	Seven Decade Counter	17.95
7209IPA	Clock Generator	3.95
7215IPG	4 Func. CMOS Stopwatch CKT	13.95
7215EV/Kit*	4 Func. Stopwatch Chip, XTL	19.95
7216AIJI	8-Digit Univ. Counter C.A.	32.00
7216CIJI	8-Digit Freq. Counter C.A.	26.95
7216DIPI	8-Digit Freq. Counter C.C.	21.95
7217IJI	4-Digit LED Up/Down Counter	12.95
7218CIJI	8-Digit Univ. LED Drive	10.95
7224IPL	LCD 4½ Digit Up Counter DRI	11.25
7226AIJL	8-Digit Univ. Counter	31.95
7226AEV/Kit*	5 Function Counter Chip, XTL	74.95
7240IJE	CMOS Bin Prog. Timer/Counter	4.95
7242IJA	CMOS Divide-by-256 RC Timer	2.05
7250IJE	CMOS BCD Prog. Timer/Counter	6.00
7260IJE	CMOS BCD Prog. Timer/Counter	5.25
7555IPA	CMOS 555 Timer (8 pin)	1.45
7556IPD	CMOS 556 Timer (14 pin)	2.20
7611BCPA	CMOS Op Amp Comparator 5MV	2.25
7612BCPA	CMOS Op Amp Ext. Cmvr. 5MV	2.95
7621BCPA	CMOS Dual Op Amp Comp. 5MV	3.95
7631CCPE	CMOS Tri Op Amp Comp. 10MV	5.35
7641CCPD	CMOS Quad Op Amp Comp. 10MV	7.50
7642CCPD	CMOS Quad Op Amp Comp. 10MV	7.50
7660CPA	Voltage Converter	2.95
8069CCQ	50ppm Band—GAP Volt Ref. Diode	2.50
8211CPA	Volt Ref/Indicator	2.50
8212CPA	Volt Ref/Indicator	2.50

* INTERSIL'S EVALUATION KITS

74C

74C00	.39			74C195	1.59
74C02	.39			74C221	1.95
74C04	.39			74C240	2.25
74C08	.39	74C106	.75	74C244	2.25
74C10	.39	74C107	1.89	74C373	2.49
74C14	.75	74C151	2.95	74C374	2.59
74C20	.39	74C154	3.95	74C901	.89
74C30	.39	74C157	2.25	74C903	1.15
74C42	1.39	74C160	1.69	74C911	10.95
74C48	1.95	74C161	1.60	74C912	10.95
74C73	.79	74C162	1.49	74C915	1.69
74C74	.79	74C163	1.69	74C917	10.95
74C85	1.95	74C164	1.59	74C922	5.49
74C86	.99	74C173	1.39	74C923	5.75
74C89	6.95	74C174	1.39	74C925	7.50
74C90	1.29	74C175	1.39	74C926	7.50
74C93	1.29	74C192	1.69	80C95	.79
74C95	1.59	74C193	1.69	80C97	.79

LINEAR

LH0002CN	6.85			NE570N	4.95
LM10CLH	4.50			LM702H	.79
LM11CLH	4.75			LM703CN	.89
LH0070-OH	6.05	LM340T-5	1.25	LM709N	.29
TL071CP	.79	LM340T-12	1.25	LM710N	.79
TL072CP	1.39	LM340T-15	1.25	LM711N	.79
TL074CN	2.49	LM341P-5	.75	LM723N	.69
LH0082CD	35.80	LM341P-12	.75	LM733N	1.00
TL082CP	1.19	LM341P-15	.75	LM739N	1.19
TL084CN	2.19	LM342P-5	.69	LM741CN	.35
LH0094CD	36.80	LM342P-12	.69	MC1741SCG	3.00
LM300H	.99	LM342P-15	.69	LM747N	.79
LM301CN	.35	LM348N	1.25	LM748N	.59
LM302H	1.95	LM350K	5.75	LM1014N	2.75
LM304H	1.95	LF351N	.60	LM1310N	1.95
LM305H	.99	LF353N	1.00	LM1458CN	.59
LM307CN	.45	LF355N	1.10	LM1488N	1.25
LM308CN	1.00	LF356N	1.10	LM1489N	1.25
LM309H	1.95	LM358N	1.00	LM1496N	1.95
LM309K	1.25	LM359N	1.79	LM1556V	1.75
LM310CN	1.75	LM370N	4.49	LM1800N	2.95
LM311H	.90	LM373N	3.25	LM1877N-9	3.25
LM312H	2.49	LM377N	2.95	LM1889N	3.20
LM317MP	1.15	LM380N	1.25	LM1896N	1.75
LM317T	1.75	LM381N	1.95	LM2002T	1.49
LM317K	3.95	LM382N	1.79	LM2877P	2.05
LM318CN	1.95	LM384N	1.95	LM2878P	2.25
LM319N	1.95	LM386N-3	1.29	LM2896P-1	2.25
LM320K-5	1.35	LM387N	1.45	LM3189N	2.95
LM320K-12	1.35	LM389N	1.35	LM3900N	.69
LM320K-15	1.35	LM392N	.69	LM3905CN	1.25
LM320T-5	1.25	LF398N	4.00	LM3909N	1.15
LM320T-12	1.25	LM399H	5.00	LM3914N	3.95
LM320T-15	1.25	TL494CN	4.49	LM3915N	3.95
LM323K-5	5.95	TL496CP	1.75	LM3916N	3.95
LM324N	.99	NE510A	6.00	RC4136N	1.25
LM329DZ	.65	NE529A	4.95	RC4151NB	3.95
LM331N	3.95	NE531H	3.95	RC4194TK	5.95
LM334Z	1.30	NE536H	6.00	RC4195TK	5.49
LM335Z	1.40	NE540H	6.00	KB4428	4.25
LM336Z	1.75	NE544N	4.95	KB4429	5.95
LM337T	1.95	NE550A	1.30	LM4500A	3.25
LM337MP	1.15	NE555V	.39	ICL8038B	4.95
LM338K	6.95	LM556N	.99	LM13080N	1.29
LM339N	.99	NE564N	3.95	LM13600N	1.49
LM340K-5	1.35	LM565N	1.25	75138N	1.95
LM340K-12	1.35	LM566CN	1.95	75450N	.89
LM340K-15	1.35	LM567V	1.25	75451CN	.39

CAPACITOR CORNER

50 VOLT CERAMIC DISC CAPACITORS

Value	1-9	10-99	100+	Value	1-9	10-99	100+
10 pf	.08	.06	.05	.001µF	.08	.06	.05
22 pf	.08	.06	.05	.0047µF	.08	.06	.05
47 pf	.08	.06	.05	.01µF	.08	.06	.05
100 pf	.08	.06	.05	.022µF	.09	.07	.06
220 pf	.08	.06	.05	.047µF	.09	.07	.06
470 pf	.08	.06	.05	.1µF	.15	.12	.10

100 VOLT MYLAR FILM CAPACITORS

.001mf	.12	.10	.07	.022mf	.13	.11	.08
.0022mf	.12	.10	.07	.047mf	.21	.17	.13
.0047mf	.12	.10	.07	.1mf	.27	.23	.17
,01mf	.12	.10	.07	.22mf	.33	.27	.22

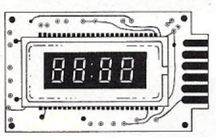
+20% DIPPED TANTALUMS (Solid) CAPACITORS

.1/35V	.39	.34	.29	1.5/35V	.41	.37	.29
.15/35V	.39	.34	.29	2.2/35V	.51	.45	.34
.22/35V	.39	.34	.29	3.3/25V	.53	.47	.37
.33/35V	.39	.34	.29	4.7/25V	.63	.56	.45
.47/35V	.39	.34	.29	6.8/25V	.79	.69	.55
.68/35V	.39	.34	.29	15/25V	1.39	1.25	.95
1.0/35V	.39	.34	.29	22/6V	.79	.69	.55

MINI. ALUMINUM ELECTROLYTIC CAPACITORS

Axial Lead				Radial Lead			
.47/50V	.16	.14	.10	.47/25V	.15	.13	.12
1.0/50V	.19	.16	.12	.47/50V	.16	.14	.13
3.3/50V	.17	.15	.11	1.0/16V	.15	.13	.12
4.7/25V	.18	.15	.11	1.0/25V	.16	.14	.13
10/25V	.18	.15	.11	1.0/50V	.17	.15	.14
10/50V	.19	.16	.12	4.7/16V	.15	.13	.12
22/25V	.19	.16	.12	4.7/25V	.16	.14	.13
22/50V	.24	.20	.18	4.7/50V	.17	.15	.14
47/25V	.25	.21	.19	10/16V	.15	.13	.12
47/50V	.29	.25	.23	10/25V	.16	.14	.13
100/25V	.28	.24	.22	10/50V	.17	.15	.14
100/50V	.41	.37	.34	47/50V	.25	.21	.19
220/25V	.39	.34	.33	100/16V	.21	.17	.14
220/50V	.49	.45	.41	100/25V	.25	.23	.21
470/25V	.54	.49	.45	100/50V	.37	.34	.31
1000/16V	.79	.69	.61	220/16V	.25	.21	.19
2200/16V	.89	.79	.69	470/25V	.35	.31	.27

National Semiconductor Clock Modules



12VDC AUTOMOTIVE/ INSTRUMENT CLOCK

APPLICATIONS:
- In-dash autoclocks
- After-market auto/ RV clocks
- Aircraft-marine clks.
- 12VDC oper. instru.
- Portable/battery powered instrumnts.

Features: Bright 0.3" green display. Internal crystal time-base. ± 0.5 sec./day accur. Auto. display brightness control logic. Display color filterable to blue, blue-green, green & yellow. Complete—just add switches and lens.

MA1003 Module $16.95

MA1023	.7" Low Cost Digital LED Clock Module	8.95
MA1026	.7" Dig. LED Alarm Clock/Thermometer	18.95
MA5036	.3" Low Cost Digital LED Clock/Timer	6.95
MA1002	.5" LED Display Dig. Clock & Xformer	9.95

National Semiconductor RAM SALE

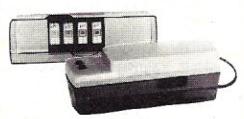
MM5290J-2 (MK4116/UPD416) . . . **$6.95 each**
16K DYNAMIC RAM (150NS)
(8 EACH $49.95) (100 EACH $550.00/lot)

MM5298J-3A . . . . **$3.25 each**
8K DYNAMIC RAM (LOW HALF OF MM5290J) 200NS
(8 EACH $23.95) (100 EACH $250.00/lot)

MM2114-3 . . . . **$5.95 each**
4K STATIC RAM (300NS)
(8 EACH $43.95) (100 EACH $450.00/lot)

MM2114L-3 . . . . **$6.25 each**
4K STATIC RAM (LOW POWER 300NS)
(8 EACH $44.95) (100 EACH $475.00/lot)

EPROM Erasing Lamp



- Erases 2708, 2716, 1702A, 5203Q, 5204Q, etc.
- Erases up to 4 chips within 20 minutes.
- Maintains constant exposure distance of one inch.
- Special conductive foam liner eliminates static build-up.
- Built-in safety lock to prevent UV exposure.
- Compact — only 7-5/8" x 2-7/8" x 2"
- Complete with holding tray for 4 chips.

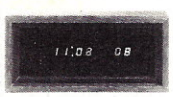
UVS-11E $79.50

Jumbo 6-Digit Clock Kit

- Four .630"ht. and two .300"ht. common anode displays
- Uses MM5314 clock chip
- Switches for hours, minutes and hold functions
- Hours easily viewable to 30 feet
- Simulated walnut case
- 115VAC operation
- 12 or 24 hour operation
- Includes all components, case and wall transformer
- Size: 6¾" x 3-1/8" x 1¾"

JE747 .... $29.95

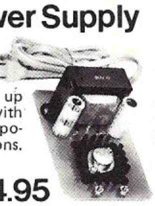
6-Digit Clock Kit



- Bright .300 ht. comm. cathode display
- Uses MM5314 clock chip
- Switches for hours, minutes and hold modes
- Hrs. easily viewable to 20 ft.
- Simulated walnut case
- 115 VAC operation
- 12 or 24 hr. operation
- Incl. all components, case & wall transformer
- Size: 6¾" x 3-1/8" x 1¾"

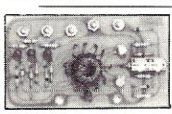
JE701 .... $19.95

Regulated Power Supply



Uses LM309K. Heat sink provided. PC board construction. Provides a solid 1 amp @ 5 volts. Can supply up to ±5V, ±9V and ±12V with JE205 Adapter. Includes components, hardware and instructions. Size: 3½" x 5" x 2"H

JE200 .... $14.95



ADAPTER BOARD —Adapts to JE200— ±5V, ±9V and ±12V

DC/DC converter with +5V input. Toriodal hi-speed switching XMFR. Short circuit protection. PC board construction. Piggy-back to JE 200 board. Size: 3½" x 2" x 9/16"H

JE205 .... $12.95

MICROPROCESSOR COMPONENTS

8080A/8080A SUPPORT DEVICES

INS8080A	CPU	6.50
DP8212	8-Bit Input/Output	3.25
DP8214	Priority Interrupt Control	5.95
DP8216	Bi-Directional Bus Driver	3.49
DP8224	Clock Generator/Driver	3.95
DP8226	Bus Driver	3.49
DP8228	System Controller/Bus Driver	4.95
DP8238	System Controller	5.95
INS8243	I/O Expander for 48 Series	9.95
INS8250	Asynchronous Comm. Element	16.95
DP8251	Prog. Comm. I/O (USART)	7.95
DP8253	Prog. Interval Timer	14.95
DP8255	Prog. Peripheral I/O (PPI)	9.95
DP8257	Prog. DMA Control	19.95
DP8259	Prog. Interrupt Control	14.95
DP8275	Prog. CRT Controller	49.95
DP8279	Prog. Keyboard/Display Interface	19.95
DP8300	Octal Bus Receiver	6.95
DP8303	System Timing Element	6.95
DP8304	8-Bit Bi-Directional Receiver	3.95
DP8307	8-Bit Bi-Directional Receiver	3.95
DP8308	8-Bit Bi-Directional Receiver	3.95

6800/6800 SUPPORT DEVICES

MC6800	MPU	14.95
MC6802CP	MPU with Clock and RAM	19.95
MC6810API	128x8 Static RAM	4.95
MC6821	Peripheral Inter. Adapt (MC6820)	7.43
MC6828	Priority Interrupt Controller	10.95
MC6830L8	1024x8-Bit ROM (MC68A30-8)	14.95
MC6850	Asynchronous Comm. Adapter	6.95
MC6852	Synchronous Serial Data Adapter	6.95
MC6860	0-600bps Digital MODEM	10.95
MC6862	2400bps Modulator	12.95
MC6880A	Quad 3-State Bus. Trans. (MC8T26)	2.25

MICROPROCESSOR CHIPS

Z80 (780C)	CPU (MK3880N)	13.95
Z80A (780-1)	CPU (MK3880N-4)	15.95
CDP1802	CPU	19.95
2650	MPU	16.95
IDM2901ADC	CPU—4-Bit Slice (Com. Temp. Grade)	19.95
MCS6502	MPU w/Clock (65K Bytes Memory)	11.95
INS8035N-6	MPU—8-Bit (6MHz)	16.95
INS8039N-6	CPU—Sgl. Chip 8-Bit (128 bytes RAM)	19.95
INS8040N-6	CPU (256 Bytes RAM)	24.95
INS8070N	CPU—64 Bytes RAM	24.95
P8085	CPU	19.95
INS8900	CPU—16-Bit	29.95
TMS9900JL	MPU—16-Bit	49.95

SHIFT REGISTERS

MM500H	Dual 25-Bit Dynamic	.50
MM503H	Dual 50-Bit Dynamic	.50
MM506H	Dual 100-Bit Static	.50
MM510H	Dual 64-Bit Accumulator	.50
MM1402	256-Bit Dynamic	3.95
MM5013	1024-Bit Dynamic/Accumulator	1.95
MM5016H	500/512-Bit Dynamic	.89
MM5034N	Octal 80-Bit	9.95
MM5035N	Octal 80-Bit	9.95
2504V (1404A)	1024-Bit Dynamic	3.95
2518N	Hex 32-Bit Static	4.95
2522V	Dual 132-Bit Static	2.95
2524V	512-Bit Dynamic	.99
2525V	1024-Bit Dynamic	2.95
2527V	Dual 256-Bit Static	2.95
2528V	Dual 250-Bit Static	4.00
2529V	Dual 240-Bit Static	4.00
2532N	Quad 80-Bit Static	2.95
3341PC	Fifo (Dual 80)	6.95

DATA ACQUISITION

AF100-1CN	Universal Active Filter 2.5%	5.95
AF121-1CJ	Touch Tone Low Pass Filter	19.95
AF122-1CJ	Touch Tone Low Pass Filter	19.95
LM308AH	Super Gain Op Amp	1.00
LM334Z	Constant Current Source	1.30
LM335Z	Temperature Transducer	1.40
LF356N	JFET Input Op Amp	1.10
LF398N	Sample & Hold Amplifiers	3.95
LM399H	Temp. Comp. Prec. Ref. (.5ppm/C°)	4.95
ADC0804LCN	8-Bit A/D Converter (1 LSB)	4.95
DAC0806LCN	8-Bit D/A Converter (0.78% Lin.)	2.25

DATA ACQUISITION (CONTINUED)

ADC0809CCN	8-Bit A/D Converter (8-Ch. Multi.)	5.25
ADC0817CCN	8-Bit A/D Converter (16-Ch. Multi.)	10.95
DAC1000LCN	10-Bit D/A Conv. Micro. Comp. (0.05%)	13.95
DAC1008LCN	10-Bit D/A Conv. Micro. Comp. (0.20%)	8.95
DAC1020LCN	10-Bit D/A Converter (0.05% Lin.)	8.49
DAC1022LCN	10-Bit D/A Converter (0.20% Lin.)	5.95
DAC1222LCN	12-Bit D/A Converter (0.20% Lin.)	9.95
CD4051N	8-Channel Multiplexer	1.19
AY-5-1013	30K BAUD UART	5.95

RAM'S

1101	256x1 Static	1.49
1103	1024x1 Dynamic	.99
2101 (8101)	256x4 Static	3.95
2102	1024x1 Static	1.75
21L02	1024x1 Static	1.95
2111 (8111)	256x4 Static	3.95
2112	256x4 Static MOS	4.95
2114	1024x4 Static 450ns	5.95
2114L	1024x4 Static 450ns Low Power	6.95
2114-3	1024x4 Static 300ns	7.49
2114L-3	1024x4 Static 300ns Low Power	7.95
2117	16,384x1 Dynamic 550ns (house marked)	4.95
MM2147J	4096x1 Fast 70ns	19.95
5101	256x4 Static	7.95
MM5261	1024x1 Dynamic Fully Decoded	.99
MM5262	2Kx1 Dynamic	.25
MM5280/2107	4096x1 Dynamic	4.95
MM5290J-2(4116)	16K Dynamic 250ns (UPD416)	6.95
MM5298J-3A	8K Dyn. 200ns (lower ½ of MM5290J)	4.95
7489	16x4 Static	1.75
UPD414/MK4027	4K Dynamic 16-pin	4.95
TMS4044-45NL	4K Static	14.95
TMS4045	1024x4 Static	14.95

PROMS/EPROMS

1702A	2K UV Erasable PROM	5.95
2708	8K EPROM	9.95
TMS2716	16K EPROM (-5V, +5V, +12V)	19.95
2716 Intel (2516) TI	16K EPROM (Single +5V)	17.95
2732 Intel (2532) TI	32K EPROM	49.95
2758	8K EPROM (450ns) (Single +5V)	7.95
5203	2048 PROM	14.95
82S23(74S188)	32x8 PROM (Open Collector)	4.95
82S115	4096 Bipolar PROM	19.95
82S123(74S288)	32x8 Tri-State Bipolar PROM	4.95
82S185	8K PROM	29.95

ROM'S

2513(2140)	Character Generator (Upper Case)	9.95
2513(3021)	Character Generator (Lower Case)	9.95
2516N	Character Generator	10.95
MM5230N	2048-Bit Read Only Memory	1.95

NMOS READ ONLY MEMORIES

MCM66710P	128x9x7 ASCII Shifted w/Greek	13.50
MCM66740P	128x9x7 Math Symbol & Pictures	13.50
MCM66750P	128x9x7 Alpha. Control Char. Gen.	13.50

MICROPROCESSOR MANUALS

M-Z80	User Manual	7.50
M-CDP1802	User Manual	7.50
M-2650	User Manual	5.00

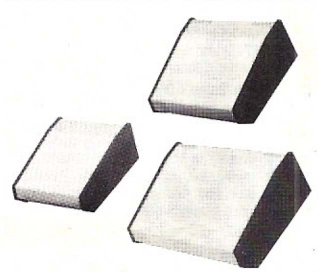
SPECIAL FUNCTION

DS0025CN	Dual MOS Clock Driver (5MZ)	3.50
DS0026CN	Dual MOS Clock Driver (5MZ)	1.95
INS1771N-1	Floppy Disc Controller	24.95
INS2651N	Communication Chip	19.95
MM58167N	Microprocessor Real Time Clock	8.95
MM58174N	Microprocessor Compatible Clock	11.95
COP402N	Microcontroller with 64-Digit RAM and Direct LED Drive	6.95
COP402MN	Microcontroller with 64-Digit RAM & Direct LED Drive w/N Buss Int.	7.49
COP470N	32-Seg. VAC Fluor. Driver (20-pin pkg.)	3.25

TELEPHONE/KEYBOARD CHIPS

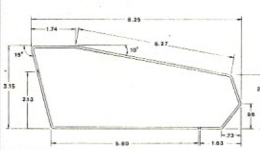
AY-5-9100	Push Button Telephone Dialer	14.95
AY-5-9200	Repertory Dialer	14.95
AY-5-9500	CMOS Clock Generator	4.95
AY-5-2376	Keyboard Encoder (88 keys)	11.95
HD0165-5	Keyboard Encoder (16 keys)	7.95
74C922	Keyboard Encoder (16 keys)	5.49
74C923	Keyboard Encoder (20 keys)	5.75
MM53190N	Push Button Pulse Dialer	7.95
MM57499N	96/144-Key Serial Keyboard Encoder	8.95

DESIGNERS' SERIES
Blank Desk-Top Electronic Enclosures



- High strength epoxy molded end pieces in mocha brown finish.
- Sliding rear/bottom panel for service and component accessibility.
- Top/bottom panels .080 thk alum. Alodine type 1200 finish (gold tint color) for best paint adhesion after modification.
- Vented top and bottom panels for cooling efficiency.
- Rigid construction provides unlimited applications.

CONSTRUCTION:
The "DTE" Blank Desk Top Electronic Enclosures are designed to blend and complement today's modern computer equipment and can be used in both industrial and home. The end pieces are precision molded with an internal slot (all around) to accept both top and bottom panels. The panels are then fastened to ¼" thick tabs inside the end pieces to provide maximum rigidity to the enclosure. For ease of equipment servicing, the rear/bottom panel slides back on slotted tracks while the rest of the enclosure remains intact. Different panel widths may be used while maintaining a common profile outline. The molded end pieces can also be painted to match any panel color scheme.



Enclosure Model No.	Panel Width	PRICE
DTE-8	8.00"	$29.95
DTE-11	10.65"	$32.95
DTE-14	14.00"	$34.95

$10.00 Min. Order — U.S. Funds Only
Calif. Residents Add 6% Sales Tax
Postage—Add 5% plus $1 Insurance (if desired)

Spec Sheets — 25¢
1981 Catalog Available — Send 41¢ stamp

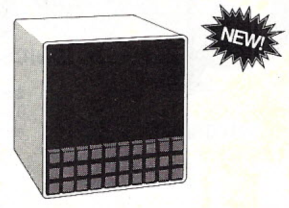
NEW 1981 CATALOG

Jameco ELECTRONICS

✓41 PHONE ORDERS WELCOME (415) 592-8097

MAIL ORDER ELECTRONICS — WORLDWIDE
1355 SHOREWAY ROAD, BELMONT, CA 94002
11/80 PRICES SUBJECT TO CHANGE

COMPUTER CUBE™



COMPUTER CRT MONITOR & ACCESSORY CASE
- One piece heavy duty molded construction
- Painted to match Apple II & III (Lt. beige, textured finish)
- Smoke colored acrylic front cover (removable)
- Built-in shelf holds CRT and allows room for 2 Apple
- disk drives below shelf
- Three 2½" holes provided in bottom of case for addition
- of fan if needed.
- Fan hole positioned above Apple motherboard location.
- Hookup cables can be run through other 2 holes.
- Case accommodates most B&W and Color uncased CRT
- monitors made by Motorola, Ball Bros., Zenith, Sanyo,
- Panasonic, Hitachi, etc. or any monitor that will fit into
- 10-3/8"H x 14½"W x 13¾"D space.
- Size: 15" x 15" x 15" O.D., 14½"H x 14½"W x 13¾" I.D.
- Weight: approximately 12 lbs.
- CRT monitor fan and disk drives not included.

CUBE-1 .... $99.95

TRS-80 16K Conversion Kit

Expand your 4K TRS-80 System to 16K.
Kit comes complete with:
* 8 each MM5290-2 (UPD416) (16K Dynamic Rams) (250NS or less)
* Documentation for conversion

TRS-16K $49.95

JE610 ASCII Encoded Keyboard Kit

The JE610 ASCII Keyboard Kit can be interfaced into most any computer system. The kit comes complete with an industrial grade keyboard switch assembly (62-keys), IC's, sockets, connector, electronic components and a double-sided printed wiring board. The keyboard assembly requires +5V @ 150mA and −12V @ 10 mA for operation. Features: 60 keys generate the 126 characters, upper and lower case ASCII set. Fully buffered. Two user-define keys provided for custom applications. Caps lock for upper-case-only alpha characters. Utilizes a 2376 (40-pin) encoder read-only memory chip. Outputs directly compatible with TTL/DTL or MOS logic arrays. Easy interfacing with a 16-pin dip or 18-pin edge connector.

JE610 (Case not included) **$79.95**
K62 (Keyboard only) $34.95

Desk-Top Enclosure for JE610 ASCII Encoded Keyboard Kit

Compact desk-top enclosure: Color-coordinated designer's case with light tan aluminum panels and molded end pieces in mocha brown. Includes mounting hardware. Size: 3½"H x 14½"W x 8¾"D.

DTE-AK $49.95
SPECIAL: JE610/DTE-AK PURCHASED TOGETHER (Value $129.90) $124.95

JE600 Hexadecimal Encoder Kit

FULL 8-BIT LATCHED OUTPUT 19-KEY KEYBOARD

The JE600 Encoder Keyboard Kit provides two separate hexadecimal digits produced from sequential key entries to allow direct programming for 8-bit microprocessor or 8-bit memory circuits. Three additional keys are provided for user operations with one having a bistable output available. The outputs are latched and monitored with 9 LED readouts. Also included is a key entry strobe. Features: Full 8-bit latched output for microprocessor use. Three user-define keys with one being bistable operation. Debounce circuit provided for all 19 keys. 9 LED readouts to verify entries. Easy interfacing with standard 16-pin IC connector. Only +5VDC required for operation.

JE600 (Case not included) **$59.95**
K19 (Keyboard only) $14.95

Desk-Top Enclosure for JE600 Hexadecimal Keyboard Kit

Compact desk-top exclosure: Color-coordinated designer's case with light tan aluminum panels and molded end pieces in mocha brown. Includes mounting hardware. Size: 3½"H x 8¼"W x 8¾"D.

DTE-HK $44.95
SPECIAL: JE600/DTE-HK PURCHASED TOGETHER (Value $104.90) $99.95

✓ Reader Service index—page 241

Quest ELECTRONICS

P.O. Box 4430S
Santa Clara, CA 95054
Will calls: 2322 Walsh Ave.
(408) 988-1640

✓44

INTEGRATED CIRCUITS

Same day shipment. First line parts only. Factory tested. Guaranteed money back. Quality IC's and other components at factory prices.

7400TTL	
7400N	.19
7402N	.20
7404N	.25
7409N	.23
7410N	.20
7414N	.70
7420N	.20
7422N	.39
7430N	.20
7442N	.58
7445N	.74
7447N	.58
7448N	.77
7450N	.20
7474N	.35
7475N	.49
7485N	.88
7489N	1.85
7490N	.43
7492N	.43
7493N	.43
7495N	.69
74100N	1.20
74107N	.35
74121N	.34
74123N	.59
74125N	.45
74145N	.77
74150N	1.20
74151N	.69
74154N	2.00
74157N	.69
74161N	.87
74162N	1.25
74163N	.87
74174N	.96
74175N	.95
74190N	1.15
74192N	.87
74193N	.85
74221N	2.75
74298N	1.65
74365N	.89
74366N	.89
74367N	.89

74LS00 TTL	
74LS00N	.35
74LS02N	.35
74LS04N	.55
74LS05N	.40
74LS08N	.45
74LS10N	.45
74LS13N	.55
74LS14N	1.25
74LS20N	.45
74LS22N	.45
74LS28N	.45
74LS30N	.45
74LS33N	.75
74LS38N	.75
74LS74N	1.25
74LS75N	1.00
74LS90N	1.00
74LS93N	1.00
74LS95N	1.10
74LS107N	.65
74LS112N	.65
74LS113N	.45
74LS132N	.89
74LS136N	.65
74LS151N	1.10
74LS155N	1.10
74LS157N	1.10
74LS162N	1.65
74LS163N	1.65
74LS174N	2.00
74LS190N	1.25
74LS221N	2.50
74LS258N	1.60
74LS367N	1.65

LINEAR	
CA3045	.90
CA3046	1.10
CA3081	1.80
CA3082	1.90
CA3089	2.95
LM301AN/AH	.35
LM305H	.87
LM307N	.35
LM308N	1.00
LM309K	1.50
LM311H/N	.90
LM317T/K	3.75
LM318	1.35
LM320K-5	1.50
LM323K-5	5.95
LM320K-12	1.50
LM320K-15	1.50
LM320T-5	1.35
LM320T-8	1.35
LM320T-12	1.35
LM320T-15	1.35
LM324N	1.40
LM339N	1.00
LM340K-5	1.35
LM340K-8	1.35
LM340K-12	1.35
LM340K-15	1.35
LM340K-24	1.35
LM340T-5	1.25
LM340T-8	1.25
LM340T-12	1.25
LM340T-15	1.25
LM340T-18	1.25
LM340T-24	1.25
LM350	7.50
LM377	3.50
LM379	5.00
LM380N	1.00
LM381	1.60
LM382	1.60
LM703H	.65
LM709H	.28
LM723H/N	.50
LM733N	.85
LM741CH	.35
LM741N	.38
LM747H/N	.75
LM748N	.35
LM1303N	1.75
LM1304	1.10
LM1305	1.27
LM1307	2.00
LM1310	2.75
LM1458	.47
LM1812	7.50
LM1889	3.00
LM2111	1.75
LM2902	2.25
LM3900N	.60
LM3905	1.75
LM3909N	.95
MC1458V	.50
NE550N	1.00
NE555V	.39
NE556A	.85
NE565A	1.00
NE566V	1.50
NE567V	1.00
NE570B	4.75
78L05	.60
78L08	.60
78M05	.85
75108	1.75
75491CN	.50
75492CN	.55
75494CN	.89

A to D CONVERTER	
8038B	4.50
8700CJ	13.95
8701CN	22.00
8750CJ	13.95
LD130	9.95
9400CJV/F	7.40
ICL7103	9.50
ICL7107	14.25

CMOS	
CD4000	.25
CD4001	.35
CD4002	.35
CD4006	1.10
CD4007	.35
CD4008	.28
CD4009	.45
CD4010	.45
CD4011	.35
CD4012	.28
CD4013	.47
CD4014	1.25
CD4015	1.00
CD4016	.55
CD4017	1.05
CD4018	.94
CD4019	.45
CD4020	1.02
CD4021	1.35
CD4022	1.10
CD4023	.28
CD4024	.75
CD4025	.28
CD4026	2.50
CD4027	.66
CD4028	.85
CD4029	1.35
CD4030	.45
CD4035	1.35
CD4040	1.35
CD4042	.85
CD4043	.85
CD4044	.85
CD4046	1.67
CD4049	.45
CD4050	.49
CD4051	1.13
CD4060	1.42
CD4066	.71
CD4068	.40
CD4069	.40
CD4070	.50
CD4071	.45
CD4072	.45
CD4073	.45
CD4075	.45
CD4076	1.65
CD4078	.40
CD4081	.35
CD4082	.35
CD4116	.47
CD4490	5.50
CD4507	1.00
CD4508	4.25
CD4510	1.02
CD4511	.94
CD4515	2.52
CD4516	1.10
CD4518	1.02
CD4520	1.02
CD4527	1.51
CD4528	.79
CD4553	3.50
CD4566	2.25
CD4583	2.35
CD4585	1.10
CD40192	3.00
74C00	.35
74C04	.40
74C10	.35
74C14	1.95
74C20	.35
74C30	.35
74C48	2.25
74C74	.85
74C76	1.75
74C90	1.75
74C93	1.75
74C154	3.00
74C160	2.00
74C175	1.35
74C192	2.25
74C221	2.50
74C905	6.00
74C906	.75
74C914	1.95
74C922	6.00
74C923	6.00
74C925	7.50
74C926	6.95
74C927	6.95

INTERFACE	
8095	.65
8096	.65
8097	.65
8098	.65
8T09	1.25
8T10	4.50
8T13	3.00
8T20	5.50
8T23	3.10
8T24	3.50
8T25	3.20
8T26	1.69
8T28	2.75
8T97	1.69
8T98	1.69

MOS/MEMORY RAM	
2101-1	2.95
2102-1	.95
2102AL-4	1.34
2102AN-2L	1.60
2104A-4	4.95
2107B-4	3.75
2111-1	3.75
2112-2	3.95
2114L 300ns	6.75
2114L 450ns	5.95
4116 200ns	7.95
8/4116 200ns	49.00
2513B	6.30
MM5262	.40
MM5280	3.00
MM5320	9.95
MM5330	5.94
PD411D-3	4.00
PD411D-4	5.00
P5101L	8.95
4200A	9.95
82S25	2.90
91L02A	1.50
HD0165-5	6.95
MM57100	4.50
GIAY38500-1	9.95
MCM66751A	9.95
9368	3.50
410D	10.00
416	16.00

CLOCKS	
MM5311	5.50
MM5312	3.90
MM5314	3.90
MM5369	2.10
MM5841	14.45
MM5865	7.95
CT7010	8.95
CT7015	8.95
MM5375AA/N	3.90
MM5375AG/N	4.90
7205	16.50
7207	7.50
7208	15.95
7209	4.95
DS0026CN	3.75
DS0056CN	3.75
MM53104	2.50

MICROPROCESSOR	
6502	10.95
6504	9.95
6522	9.95
6800	6.95
6802	11.95
6820	4.95
6850	5.95
8080A	5.95
8085	12.95
8086	75.00
Z80	9.95
Z80A	11.95
8212	2.90
8214	3.95
8216	2.90
8224	3.45
8228	4.95
8251	6.95
8253	15.00
8255	5.75
8257	10.95
8259	14.95
1802CP plas.	13.95
1802DP plas.	17.95
1861P	11.50
CDP1802CD	28.95
CDP1802D	35.00
CDP1861	15.95

UART/FIFO	
AY5-1013	5.50
AY5-1014	7.50
3341	6.95

PROM	
1702A	4.95
2513B upper case	8.75
2708	7.75
2716T1	18.00
2716 Intel	23.00
8/2716 Intel	160.00
2732	65.00
2758	22.50
8741A	60.00
8748	65.00
8748-8	60.00
8755A	55.00
N82S23	2.95
N82S123	6.50
N82S126	3.75
N82S129	8.50
N82S131	8.50
N82S136	8.75
N82S137	8.75
DM8577	2.90
8223	2.90

CONNECTORS	
30 pin edge	2.50
44 pin edge	2.75
100 pin edge	4.50
100 pin edge WW	5.25

IC SOCKETS
Solder Tin Low Profile

PIN	1 UP	PIN	1UP
8	.15	22	.30
14	.14	24	.35
16	.16	28	.42
18	.27	36	.58
20	.29	40	.57

2 level 14 pin ww .20

WIRE WRAP LEVEL 3

PIN		PIN	
14	.32	24	.86
16	.33	28	1.00
18	.57	40	1.23

CRYSTALS	
1 MHz	4.50
2 MHz	4.50
4 MHz	4.25
5 MHz	4.25
10 MHz	4.25
18 MHz	3.90
20 MHz	3.90
32 MHz	3.90
32768 Hz	4.00
1.8432 MHz	4.50
3.5795 MHz	1.20
2.0100 MHz	1.95
2.097152 MHz	4.50
2.4576 MHz	4.50
3.2768 MHz	4.50
5.0688 MHz	4.50
5.185 MHz	4.50
5.7143 MHz	4.50
6.5536 MHz	4.50
14.31818 MHz	4.25
18.432 MHz	4.50
22.1184 MHz	4.50

KEYBOARD ENCODERS	
AY5-2376	$12.50
AY5-3600	17.95
AY5-9100	10.50
AY5-9200	16.50
74C922	5.50
74C923	5.50
HD0165-5	6.95
AY5-9400	10.50

D Connectors RS232	
DB25P	3.62
DB25S	5.20
Cover	1.67
DE9S	1.95
DA15P	2.10
DA15S	3.10
Complete Set	9.50

Hickok 3½ Digit LED multimeter	**74.95**
Stopwatch Kit	**26.95**
Auto Clock Kit	**17.95**
Digital Clock Kit	**14.95**
8K/16K Eprom Kit (less PROMS)	$89.00
Motherboard	$39.00
Extender Board	$15.00

RESISTORS ¼ watt 5%
10 per type .03
25 per type .025
100 per type .015
1000 per type .012
350 piece pack
5 per type 6.75
½ watt 5% per type .05

Televideo Terminal
Model 912 $845.00
Model 920 $945.00

Tiny Basic Experimentors Kit **$10.00**

BSR Controller $39.95
Connect your computer to the BSR Home Control System. Computer controlled ultrasonic transmitter for your BSR. Software for 1802 user.

KEYBOARDS	
56 key ASCII keyboard kit	$67.50
Fully assembled	77.50
53 key ASCII keyboard kit	60.00
Fully assembled	70.00
Enclosure Plastic	14.95
Metal Enclosure	29.95

LEDS	
Red T018	.15
Green, Yellow T018	.20
Jumbo Red	.20
Green, Orange, Yellow Jumbo	.25
Cliplite LED Mounting Clips	8/$1.25

(specify red, amber, green, yellow, clear)

CONTINENTAL SPECIALTIES in stock
Complete line of breadboard test equip.
MAX-100 8 digit Freq. Ctr. $128.95

OK WIRE WRAP TOOLS in stock
Portable Multimeter $18.00

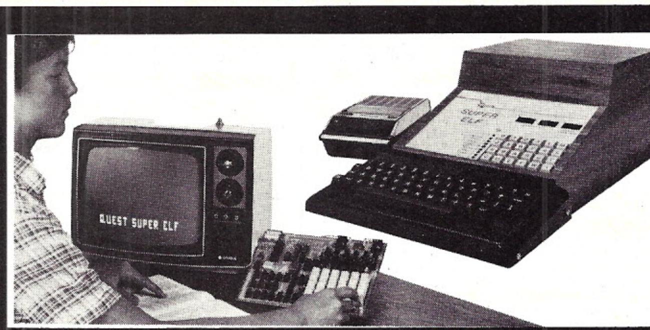
Complete line of AP Products in stock.

SPECIAL PRODUCTS	
MM5865 Stopwatch Timer with 10 pg. spec.	9.00
PC board	7.50
Switches Mom. Pushbutton	.27
3 pos. slide	.25
Encoder HD0165-5	6.95
Paratronics 100A Logic Analyzer	**$425.00**
Model 10 Trigger Expander Kit	$229.00
Model 150 Bus Grabber Kit	$369.00
Clock Calendar Kit	**$23.95**
2.5 MHz Frequency Counter Kit	**$37.50**
30 MHz Frequency Counter Kit	**$47.75**

TRANSFORMERS	
6V 300 ma	3.25
12 Volt 300 ma transformer	1.25
12.6V CT 600 ma	3.75
12V 250 ma wall plug	2.95
12V CT 250 ma wall plug	3.50
24V CT 400 ma	3.95
10V 1.2 amp wall plug	4.85
12V 6 amp	12.95
12V 500 ma wall plug	4.75
12V 1 amp wall plug	6.50
10/15 VAC 8/16 VA wall plug	9.75

DISPLAY LEDS			
MAN1	CA	.270	2.90
MAN3	CC	.125	.39
MAN72/74	CA/CA	.300	1.00
DL704	CC	.300	1.25
DL707/DL707R	CA	.300	1.00
DL727/728	CA/CC	.500	1.90
DL747/750	CA/CC	.600	1.95
DL750	CC	.600	1.95
FND359	CC	.357	.70
FND500/507	CC/CA	.500	1.35
FND503/510	CC/CA	.500	.90
FND800/807	CC/CA	.800	2.20
3 digit Bubble			.60
4 digit Bubble			.80
DG8 Fluorescent			1.75
DG10 Fluorescent			1.75
10 digit display			1.25
7520 Clairex photocells			.39
TIL311 Hex			9.50
MAN3640	CC	.30	1.10
MAN4610	CA	.40	1.20
MAN4640	CC	.40	1.20
MAN4710	CA	.40	.95
MAN4740	CC	.40	1.20
MAN6640	CC	.56	2.95
MAN6710	CA	.60	1.35
MAN6740	CC	.60	1.35

MA1002A	**8.95**
MA1002E	**8.95**
MA1012A	**8.95**
102P3 transformer	**2.25**
MA1012A Transformer	**2.25**



RCA Cosmac 1802 Super Elf Computer $106.95

Compare features before you decide to buy any other computer. There is no other computer on the market today that has all the desirable benefits of the **Super Elf** for so little money. The Super Elf is a small single board computer that does many **big** things. It is an excellent computer for training and for learning programming with its machine language and yet it is easily **expanded** with **additional memory, Full Basic, ASCII Keyboards, video character generation, etc.**

Before you buy another small computer, see if it includes the following features: ROM monitor; State and Mode displays; Single step; Optional address displays; Power Supply; Audio Amplifier and Speaker; Fully socketed for all IC's; Real cost of in warranty repairs; Full documentation.

The **Super Elf** includes a **ROM** monitor for program loading, editing and execution with **SINGLE STEP for program debugging** which is not included in others at the same price. With **SINGLE STEP** you can see the microprocessor chip operating with the **unique Quest** address and data bus displays **before, during** and **after** executing instructions. Also, CPU mode and instruction cycle are decoded and displayed on 8 LED indicators.

An RCA 1861 **video graphics** chip allows you to connect to your own TV with an inexpensive video modulator to do graphics and games. There is a **speaker system** included for writing your own music or using many music programs already written. The speaker amplifier may also be used to drive relays for control purposes.

A 24 key **HEX keyboard** includes 16 HEX keys plus **load, reset, run, wait, input, memory protect, monitor select** and **single step**. Large, on board displays provide output and optional **high and low address**. There is a 44 pin standard connector slot for PC cards and a 50 pin connector slot for the Quest Super Expansion Board. Power supply and sockets for **all** IC's are included in the price plus a detailed 127 pg. instruction manual which now includes over 40 pgs. of software info. including a series of lessons to help get you started and a music program and graphics target game. Many schools and universities are using the Super Elf as a course of study. OEM's use it for training and R&D.

Remember, other computers only offer Super Elf features at additional cost or not at all. **Compare before you buy. Super Elf Kit $106.95, High address option $8.95, Low address option $9.95. Custom Cabinet** with drilled and labelled plexiglass front panel **$24.95.** All metal Expansion Cabinet, painted and silk screened, with room for 5 S-100 boards and power supply **$57.00. NiCad Battery Memory Saver Kit $6.95.** All kits and options also completely assembled and tested.

Questdata, a software publication for 1802 computer users is available by subscription for $12.00 per 12 issues. Single issues $1.50. Issues 1-12 bound $16.50.

Tiny Basic Cassette **$10.00,** on ROM **$38.00,** original Elf kit board **$14.95. 1802 software;** Moews Video Graphics **$3.50.** Games and Music **$3.00,** Chip 8 Interpreter **$5.50.**

Super Expansion Board with Cassette Interface $89.95

This is truly an astounding value! This board has been designed to allow **you** to decide how you want it optioned. **The Super Expansion Board comes with 4K of low power RAM** fully addressable anywhere in 64K with built-in memory protect and a **cassette interface**. Provisions have been made for all other options on the same board and it fits neatly into the hardwood cabinet alongside the **Super Elf**. The board includes slots for up to 6K of **EPROM** (2708, 2758, 2716 or TI 2716) and is **fully socketed.** EPROM can be used for the monitor and Tiny Basic or other purposes.

A IK Super ROM Monitor $19.95 is available as an on board option in 2708 EPROM which has been preprogrammed with a program loader/editor and error checking multi file cassette read/write software, (relocatable cassette file) another exclusive from Quest. It includes register save and readout, block move capability and video graphics driver with blinking cursor. Break points can be used with the register save feature to isolate program bugs quickly, then follow with single step. If you have the **Super Expansion Board** and **Super Monitor** the monitor is up and running at the push of a button.

Other on board options include **Parallel Input and Output Ports** with **full handshake.** They allow easy connection of an ASCII keyboard to the input port. **RS 232 and 20 ma Current Loop** for teletype or other device are on board and if you need more memory there are two **S-100** slots for static RAM or video boards. Also a 1K Super Monitor version 2 with video driver for full capability display with Tiny Basic and a video interface board. **Parallel I/O Ports $9.85, RS 232 $4.50, TTY 20 ma I/F $1.95, S-100 $4.50. A 50 pin connector set** with ribbon cable is available at $15.25 for easy connection between the **Super Elf** and the **Super Expansion Board.**

Power Supply Kit for the complete system (see Multi-volt Power Supply).

Announcing Quest Super Basic— *SECOND GENERATION*

A new enhanced version of **Super Basic** now available. Quest was the first company worldwide to ship a full size Basic for 1802 Systems. A complete function **Super Basic** by **Ron Cenker** including floating point capability with scientific notation (number range $\pm .17E^{38}$), 32 b1t integer ± 2 billion; multi dim arrays, string arrays; string manipulation; cassette I/O; save and load, basic, data and machine language programs; and over 75 statements, functions and operations.

Enhancements include increased speed, built-in provisions for Stringy Floppy, Floppy Disc, Printer Driver, I/O, user definable command library and statement renumbering.

Easily adaptable to most 1802 systems. Requires 16K RAM minimum for Basic and user programs. Source listing for both Serial and Parallel I/O included.

Super Basic on Cassette $40.00.

PROM Eraser

assembled. 25 PROM capacity **$37.50** (with timer **$69.50**). 6 PROM capacity OSHA/UL version **$69.50** (with timer **$94.50**).

Z80 Microcomputer

16 bit I/O, 2 MHz clock, 2K RAM, ROM Breadboard space. Excellent for control. Bare Board **$28.50.** Full Kit **$99.00. Monitor $20.00.** Power Supply Kit **$35.00. Tiny Basic $30.00**

S-100 Computer Boards

8K Static Godbout Econo IIA Kit	145.00
16K Static Godbout Econo XIV Kit	285.00
24K Static Godbout Econo VIIA-24 Kit	435.00
32K Static Godbout Econo X-32 Kit	575.00
16K Dynamic RAM Kit	199.00
32K Dynamic RAM Kit	310.00
64K Dynamic RAM Kit	470.00
Video Interface Kit	**$135.00**

80 IC Update Master Manual $55.00

Comp. IC data selector, 2700 pg. master reference guide. Over 51,000 cross references. Free update service through 1980. Domestic postage $3.50.

Modem Kit $60.00

State of the art, orig., answer. No tuning necessary. 103 compatible 300 baud. Inexpensive acoustic coupler plans included.

LRC 7000+Printer $389.00

40/20 column dot matrix impact, std. paper. Interface all personal computers.

64/40/32/20 version **$405.00.** Optional cables available.

LRC 7000 printer interface cable for Super Elf with software $26.00

NiCad Battery Fixer/Charger Kit

Opens shorted cells that won't hold a charge and then charges them up, all in one kit w/full parts and instructions. **$7.25**

Rockwell AIM 65 Computer

6502 based single board with full ASCII keyboard and 20 column thermal printer. 20 char. alphanumeric display, ROM monitor, fully expandable. **$375.00.** 4K version **$450.00.** 4K Assembler **$85.00,** 8K Basic Interpreter **$100.00.**

Special small power supply for AIM65 assem. in frame **$54.00.** Complete AIM65 in thin briefcase with power supply **$499.00.** Molded plastic enclosure to fit both AIM65 and power supply **$47.50.** Special Package Price: 4K AIM, 8K Basic, power supply, cabinet **$599.00**

AIM65/KIM/VIM/Super Elf 44 pin expansion board; 3 female and 1 male bus. Board plus 3 connectors **$22.95.**

60 Hz Crystal Time Base Kit $4.40

Converts digital clocks from AC line frequency to crystal time base. Outstanding accuracy.

Video Modulator Kit $8.95

Convert TV set into a high quality monitor w/o affecting usage. Comp. kit w/full instruc.

Multi-volt Computer Power Supply

8v 5 amp, ±18v .5 amp, 5v 1.5 amp, −5v .5 amp, 12v .5 amp, −12v option. ±5v, ±12v are regulated. Basic Kit **$29.95.** Kit with chassis and all hardware **$43.95.** Add $4.00 shipping. Kit of hardware **$14.00.** Woodgrain case **$10.00.** $1.50 shipping.

Gremlin Color Video Kit $69.95

32 x 16 alpha/numerics and graphics; up to 8 colors with 6847 chip; 1K RAM at E000. Plugs into Super Elf 44 pin bus. No high res. graphics. On board RF Modulator Kit **$4.95**

Elf II Adapter Kit $24.95

Plugs into Elf II providing Super Elf 44 and 50 pin plus S-100 bus expansion. (With Super Expansion). High and low address displays, state and mode LED's optional **$18.00.**

1802 16K Dynamic RAM Kit $149.00

Expandable to 32K. Hidden refresh w/clocks up to 4 MHz w/no wait states. Addl. 16K RAM **$63.00**

Super Elf 44 pin expansion board; 3 female and 1 male bus. Board plus 3 connectors **$22.95**

Tiny Basic Extended on Cassette $15.00
(added commands include Stringy, Array, Cassette I/O etc.)

S-100 4-Slot Expansion $ 9.95
Super Monitor VI.I Source Listing $15.00

Super Color S-100 Video Kit $129.95

Expandable to 256 x 192 high resolution color graphics. 6847 with all display modes computer controlled. Memory mapped. 1K RAM expandable to 6K. S-100 bus 1802, 8080, 8085, Z80 etc.

Editor Assembler $25.00
(Requires minimum of 4K for E/A plus user source)

1802 Tiny Basic Source listing $19.00
Super Monitor V2.0/2.1 Source Listing $20.00

TERMS: $5.00 min. order U.S. Funds. Calif residents add 6% tax.
$10.00 min. order BankAmericard and Master Charge and COD. $1.00 insurance optional.
Shipping charges will be added on charge cards.

FREE: Send for your copy of our NEW 1980 QUEST CATALOG. Include 48¢ stamp.

New Summer 1980
SEND FOR OUR FREE CATALOG

CompuMart lets you put 'em on trial

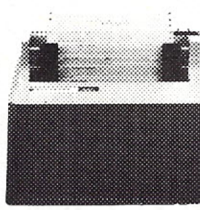
CompuMart has been selling computers by mail since 1971. Our thousands of satisfied customers rely on CompuMart for services not generally available from the others. Namely:

- Product Selection/Each product advertised by CompuMart has been evaluated by our in-house staff for best price, performance, and supplier reliability
- Return Privilege/After receipt of our products, you are protected by CompuMart's exclusive, 10-day return priviledge- good for all products except software.
- Support/Our Customer Service Dept. and expert technicians are always there to assist you by phone or at CompuMart's outlets. Our knowledgeable phone sales force can provide you with detailed information and complete product specifications.
- Phone Ordering/For added convenience, CompuMart maintains a toll-free ordering number. 1-800-343-5504.
- Phones open M T W Th F 8:30 a.m. - 7:00 p.m. Sat. 11:00 a.m. - 4:00 p.m.

with 10 day free return

Printers

The Paper Tiger Printer From Integral Data



Uses standard ⅜ inch roll paper and ribbon
40 characters per line
Speed: 40 characters per second
UL approved
High resolution dot matrix impact printer

IDS Paper Tiger Printer **$995**
IDS Graphics Paper Tiger Printer **$1,094**

FREE Cable with your Paper Tiger.

NEW! From Integral Data. The IDS 460.

We saw this new desktop printer at the NCC 80 and when we saw its features: Correspondence quality printing, High-resolution graphics capability, programmable print control functions, and automatic text justification—we knew that we had to offer this printer to our cost/features conscious customers $1,295

The Omni 810 Printer from Texas Instruments

TI Omni 820 Receive-Only (RO) Package. Includes machine-mounted paper tray and cable. A compressed print option and device forms control are standard features **$2,155**

TI Omni 820 Keyboard Send Receive (KSR) Package Comes with full ASCII Keyboard with numeric Key- pad and an EIA cable with autospeed select. **$2,395**

CENTRONICS PRINTERS

New! The incredible Model 737- Correspondence and Draft Quality Printing for Under $1,000. This is the first printer in its class to offer print quality suitable for text processing, plus the performance and application flexibility required for data processing.

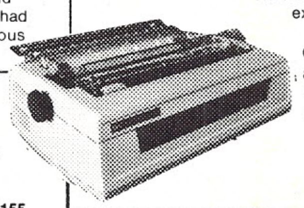
737-1 (Parallel Interface)— **$899**
737-3 (Serial Interface) **$1,045**

Tractor Feed Printer- Centronics' Most Popular Model. Perfect for the needs of a small business sytesm. Recommended by Apple and Radio Shack. **$1,079**

FREE Cable with purchase of any Centronics printer.

NEC The First Name in Letter Quality Printers.

CompuMart offers beautiful print quality with NEC Spinwriter terminals. The Spinwriters, both KSR and RO versions, give unsurpassed hard copy output. CompuMart offers a complete range of NEC Spin- writers—Call our expert salesforce



Compumart will throw in a **FREE** fabric ribbon and Currier 72 element when you buy a NEC from us.

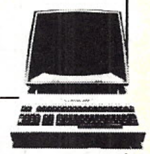
COMPUMART NOW OFFERS THE ENTIRE DEC LSI-11 PRODUCT LINE. CALL FOR PRICES & DELIVERY.

Terminals

We've got the following Lear Siegler Terminals In Stock at prices too low to print—**Call** for quotes.

ADM-3A Industries favorite dumb terminal for some very smart reasons.
ADM-3A.+ New from Lear Siegler. CALL!
ADM-31. The terminal that's too smart to be considered dumb.
ADM-42. Available with keyboard semi-intelligent terminal offering tremendous user flexibility. The optional configurations are amazing.

Call for details.



HAZELTINE TERMINALS AT SPECTACULAR SAVINGS!

Hazeltine 1410.	List $850	**CompuMart $749**
Hazeltine 1420.	List $995	**CompuMart $895**
Hazeltine 1500.	List $1095	**CompuMart $995**
Hazeltine 1510.	List $1395	**CompuMart $1325**
Hazeltine 1520.	List $1585	**CompuMart $1485**
Hazeltine 1552.	List $1395	**CompuMart $1295**

Call CompuMart for complete specs and quantity discounts.

Calculators

A CALCULATOR, A SYSTEM, A WHOLE NEW STANDARD.

HEWLETT-PACKARD'S HP-41C

HP-41C Calculator **$288.00**
The System
Memory Modules. For storing programs or up to 2,000 lines of program memory **$45.00**
"Extra Smart" Card Reader. Records programs and data back onto blank mag-cards **$199.00**
The Printer. Upper and Lower case, High resolution plotting, Portable Thermal operation **$355.00**
Application Modules .... **$45.00 EACH**
Standard pac:
Statistics,
Math,
Financial &
Surveying

1/2 off Financial Decisions PAC with purchase of HP-41

NOVATION CAT™

ACCOUSTIC MODEM

- Answer Originate
- 300 Baud
- Bell 108
- Low Profile Design

Looks good, works great! $169.00

Texas Instruments TI-99/4 Home Computer

Save $300 on this 16-Bit computer with monitor

TI-99/4 w/Monitor $1250
TI-99/4 w/o Monitor $ 950

Monitors

NEW FROM SANYO— Four Great Monitors at Low CompuMart Prices.

Sanyo's new line of CRT data display monitors are specifically designed for the display of alphanumeric or graphic data.

9″ Sanyo Monitor	$179
12″ Sanyo Monitor	$289
12″ Sanyo Monitor with green screen	$299
13″ Sanyo Color Display Monitor	$495

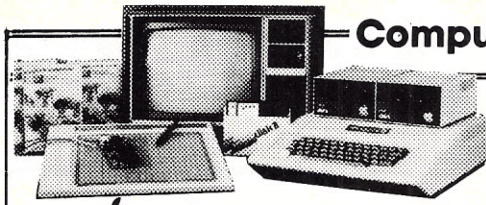
Free 6' Cable with purchase of any Sanyo

EXCLUSIVE from CompuMart!

Special Offer. Zenith Color Video Monitor for $379!

The perfect monitor for Apple, Atari and Texas Instruments owners.

Computers



apple computer

We carry the most complete inventory of Apple computers, peripherals, and software. CALL!

Our Christmas Apple Special: Save over $250 on our most popular Apple System. System includes a 48K Apple II, Apple Disk & Controller, and a Sup R Mod RF Modulator.

List: $2,020

Compumart Sale Price: $1,769

New from Apple for the Apple II:

DOS 3-3 Convert disks to 16 sector format for 23% more storage and faster access	$60
Apple Plot. The perfect graphic complement for Visicalc.	$70
Dow Jones News & Quotes	$95
Adventure (Uses 48K)	$35
DOS Tool Kit	$75
Apple Fortran	$200
Silentype Printer w/Xface	$595
Visicalc	$149
Tax Planner	$120
From Symtech & Info Unlimited	
Super Sound Generator (mono) **$159** (stereo)	$259
Light Pen	$249
X-10 Controller (plugs into paddle port)	$49
Apple Sync Controller	$49
From **Personal Software**	
Visicalc	$149
Desk top plan	$99

New from **Videx!** — Video Term
80 Col. x 24 line
7 x 9 matrix, plug in compatible board for the Apple II. Price **$325** without graphics EPROM. With graphics EPROM **$350.**

New from **MUSE**

The Voice	$39.95
Super Text	$99.00
Address Book	$49.95

Mountain Hardware — Expansion accessories for your Apple

Introl/X-10 System	$289
Super Talker	$299
The Music System	$545
ROM plus board w/keyboard filter	$199
Clock Calendar	$280
16 Channel A to D Converter	$350
Apple Expansion Chassis	$650
ROM Writer	$175
Miscellaneous Apple II Accessories:	
Easy Writer (80 col. need Videx)	$249
Easy Mover	$ 49
Easy Mailer	$ 69
Dysan Diskettes	ea. $ 5
S.S.M. Serial & Parallel Apple Interface	$225
ABT's Numeric Key Pad	$110

Preview of the Apple III

THANKSGIVING SPECIAL

Buy **$1,000** worth of merchandise from this ad, including at our special sale prices and deals, and we'll ship you a Texas Instrument's Speak & SpellTM with your order. This is TI's famous talking and learning aid with the electronic voice and brain. It's an incredible electronic learning aid for children aged 7 and up. A $69.95 Value Yours **FREE** with **$1,000** purchase.

SUPER SALE PRICED TO MOVE!

We want to move our inventory of Zenith Z-89 and Exidy Computers. To do it, we've priced them so low you'll have to call us for prices. If you want either of these two great computers, call us now while supplies last for the best prices you'll find anywhere. Good selection of Zenith and Exidy peripherals as well.

COMMODORE

Buy direct from the biggest — Compumart has delivered more Commodore computers in the U.S. than any other dealer. We were Commodore's first dealer and carry everything Commodore manufactures. In stock for immediate delivery! Call us now for low prices and special deals.

NEW FOR PET:

Visicalc (Need 32K and a disk drive) **$199**

Word Pro 1, **$29.95** • Word Pro 2, **$99.95** •

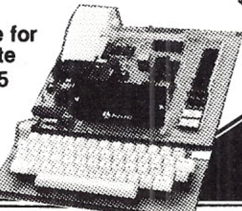
Word Pro 3, **$199.95** • Word Pro 4, **$299.95** •

ROCKWELL AIM 65

The single board development system that's perfect in the classroom or lab.

Our AIM System includes: 4K AIM with BASIC interpretor assembler, Power Supply, Cassette recorder & Enclosure	**$799.**
4K AIM — 65	**$499**
PL65 High Level Language	**$125**
Paper for the AIM (roll)	**$ 2.50**
Rockwell's 4-slot Motherboard (SALE)	**$175**
CompuMart's Microflex 65 System for your AIM Includes: Adapter Buffer Module w/ 4-slot module stack, 8K RAM module, 16K PROM/ROM module, Asynchronous communications Interface, & Power Supply	**$1,299**

Call or write for our complete Microflex 65 brochure



CompuMart STOCKS THE COMPLETE LINE OF MATROX PRODUCTS. CALL FOR SPECS.

Introducing the HP-85 $3,250

Hewlett-Packard's Personal Computer for Industry. This extremely portable computer features extended BASIC to solve your problems quickly and efficiently along with an advanced graphics system to enhance communication.

NEW from Hewlett-Packard
HP 82900-Series Flexible Disk Drives for the HP-85

These 4 new Flexible Disk Drives provide fast on-line storage using flexible disks

- *HP 82901M. Supplies approx. 540K bytes of on-line storage. **$2,500**
- *HP 82902M. Approx. 270K bytes of on-line storage. **$1,500**
- *HP 82901S. Supplies an additional 540K bytes when connected to an HP 82901M or an HP 82902M. **$2,200**
- *HP 829025. Supplies an additional 270K bytes when connected to an HP 82901M or an HP 82902M. **$1,300**

The Hewlett-Packard 7225A. High Quality/Low Cost Graphics Plotter. **$2,050**

Call our expert sales force for complete product specifications.

COMPUMART'S EXCLUSIVE ATARI SPECIALS. *(Pick one)*

3 Ways to save when you buy the Atari 800 from us.

1) **Free** 8K of memory with purchase. (So your Atari will come to you with 24K.)

2) **Free** 410 program recorder with purchase ($89.95 value).

3) $100 off Atari Disk Drive purchase.

ATARI 800 Personal Computer System —

Comes with 800 Operators Manual, 16K RAM Memory module, 10 K ROM Operating System, power supply, TV Switch Box. **$1080.00**

PERIPHERALS

Atari 410 Program Recorder (FREE w/purchase of Atari 800)	**$ 89.95**
Atari 810 Disk Drive ($100 off with purchase)	**699.95**
New Dual Disk double density	**1499.95**
825 Printer (Centronics 737)	**995.00**
RS232 Interface w/Cable	**249.95**
NEW! Light Pens	**74.95**

ATARI

Call for New Software

IMPORTANT ORDERING INFORMATION All orders must include 4% shipping and handling. Mass. residents add 5% sales tax, Michigan residents 4% for sales tax.

TO ORDER CALL: 800-343-5504
In Mass. call 1-617-491-2700

Phones open from 8:30 a.m. to 7:00 p.m., Mon.-Fri.; 11:00 a.m. - 4:00 p.m. Sat. P.O.'s accepted from Dun & Bradstreet rated companies - shipment contingent upon receipt of signed purchase order. Sale prices valid for month of magazine date only - all prices subject to change without notice. Our Ann Arbor retail store is open 11:00 a.m. to 7:00 p.m. Tues.-Fri., 10:00 a.m. to 5:00 p.m. Saturdays. Stop by and visit.

COMPUMART

43

270 THIRD ST., P.O. BOX 568, DEPT. 131, CAMBRIDGE, MA 02139

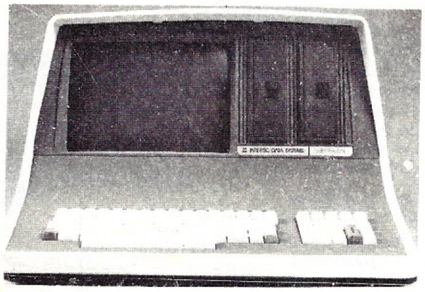
master charge VISA

Member Computer Dealers Assoc.

We've had a reputation for dependability since 1971

Computers, Disk Systems

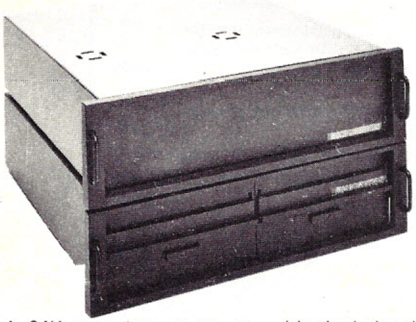
SUPERBRAIN®
By INTERTEC



32K or 64K (Double or Quad Density units available). Uses two Z-80 CPU's Commercial-type terminal with 12" monitor. Dual double density minifloppies. Over 350 kilobytes of storage (twice that with quad density drives). Two serial RS232 ports, I/O ports standard. Expandable with optional S-100 S-100 interface. Comes with CP/M™ 2.2 operating system. MiniMicroMart includes BASIC interpreter and can supply a wide range of CP/M Development and Application software.

w/32K Double Density, List $2995 . **$2685**
w/64K Double Density, List $3345 . . . . . **$2883**
w/64K Quad Density, List $3995 . . . . . . **$3595**
W/64K Quad — MiniMicroMart Upgrade Special. **$3395**

MICROMATION



A 64K complete computer with dual density 8" floppies (1 megabyte). Rack or vertical mounting. Systems with double-sided drives, hard disks, and multi-user (MP/M).

Z+ 100 64K RAM, Computer, $2495. . **$2099**
Z+120 Includes two 8" disks, $4995 . . . **$4199**

"Z" system features new distributed processing multi-user concept with one Z-80 per user, with Z-80 for MP/M (Master Satellite concept).

AS LOW AS $11,899!

SD SYSTEMS

SDS-100, w/32K RAM, $6995 . . . . . **$5945**
SDS-200, List $8995 . . . . **$7645**

RADIO SHACK TRS-80™

10% OFF!

INTERSYSTEMS
formerly ITHACA AUDIO

DPS-1, List $1795

Call for Price!

The new Series II CPU Board features a 4 MHz Z-80A CPU and a full-feature front panel. 20-slot actively terminated motherboard, with 25 amp power supply (50/60 Hz operation, incl. 68 cfm fan).

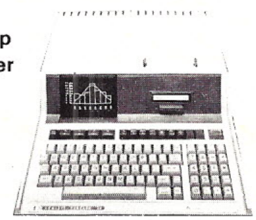
COMPLETE SYSTEM with InterSystem 64K RAM, I/O Board w/priority interrupt and double density disk controller board. Full 1-year warranty, List $3595

Call for Price!

HEWLETT-PACKARD
HP-85A

Desk-Top Computer

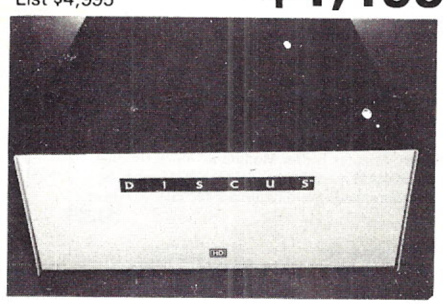
Call for Price



MORROW
THINKER TOYS®
DISCUS M26™

26 megabytes of formatted storage
List $4,995

$4,199



THINKER TOYS®
DISK SYSTEMS

Now includes CP/M® 2.2

Discus 2D, List $1199. . . . . . **$1019**
Discus 2D, dual-drive, List $1994 . . . . . . **$1694**
Discus 2+2, Assem., List $1549. . . . . . . **$1319**
Dual Discus 2+2, Assem., $2748. . . . . . **$2335**

All Morrow systems now include CP/M® 2.2

NORTH STAR MDS-A
Double Density Mini Floppy Disk System

Double Density, Kit
List $799 **OUR PRICE $669**
Assembled and Tested. **$719**
Quad Version, Kit, List. **$836**
Assembled, List $1099. . . . . **$896**

Above MDS-A units do not include cabinet or power supply.
Shipping and Insurance: Add $7.50.

NEW! CROMIX FROM CROMEMCO
A New UNIX Like Disk Operating System,

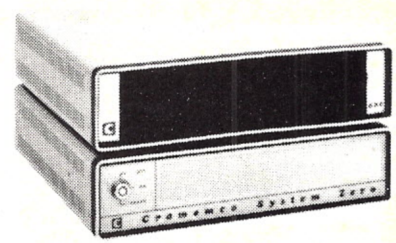
With true multi-user, multi-tasking capabilities

List $295 . . . . **OUR PRICE $249**

NEW! DOUBLE DENSITY CONTROLLER BOARD
FROM CROMEMCO

With built-in diagnostics

16 FDC Controller, List $595 **OUR PRICE $505**



NEW! CROMEMCO SYSTEM ZERO

List $995 . . . . **OUR PRICE $849**

NEW! CROMEMCO SYSTEM ZERO/D

A complete 64K Computer with Double Density Disk Controller . . . . . . List **$2995**

OUR PRICE $2545

Companion Disk drive for above — **Quad Density** — Total of 780 Kilobytes of storage on the two drives. . . . . . **List $1295**

OUR PRICE $1099

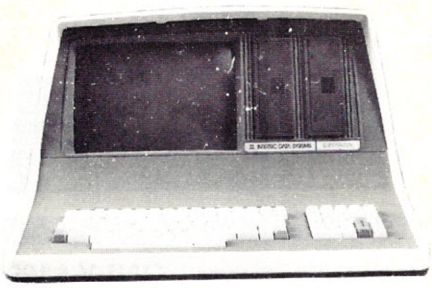
Only **$3644** for a complete 64K Disk System

MORROW
Discus 2D's IN STOCK

✓226

Try to beat our prices!

SUPERBRAIN® by Intertec



Self-contained computer with dual disks and two RS232C ports. Complete with CP/M® 2.2 and BASIC.

32K Double Density, List $2995 . **$2685**
64K Double Density, List $3345 .... $2883
64K MiniMicroMart-upgraded to Quad Density .... ***SPECIAL $3395***

VIDEO TERMINALS

***NEW* EMULATOR (Intertec)**, List $895 $ 749
NEW INTERTUBE III List $895 ... ONLY $ 749
SOROC 120, List $995 ...... ***SPECIAL $ 729***
IQ140, List $1495 .... ***SPECIAL $1149***
PERKIN-ELMER 550, List $997 ...... $ 799
with anti-glare screen, $1027 .... $ 829
HAZELTINE 1410, List $900 .... $ 749
1420 .... $ 849
1500, List $1225 .... $ 879
1510, List $1395 .... $1089
1520, List $1650 .... $1389
ADDS R-20, List $995 .... $ 945
LEAR SIEGLER ADM3A, Assembled .. $ 849
TELEVIDEO 912C, List $950 .... $ 789
920C, List $1030 .... $ 849

PRINTERS

ANADEX DP-8000 .... $ 849
DP-9500, List $1650 .... $1399
DP-9501, List $1650 .... **$1399**
PAPER TIGER IDS-440, List $995 .... $ 895
w/graphics op., incl. buffer, $1195 ... $ 989
NEW IDS PAPERTIGER 460 . List 1295 **$1149**
NEW IDS PAPERTIGER 460G List $1394 **$1199**
NEC Spinwriters .... Call for Price
TELETYPE 43 KSR $1087
CENTRONICS
730-1 parallel interface .... ***NEW LOW $679***
737 parallel interface .. ***SUPER VALUE $849***
779 w/Tractor, List $1350 .... $1049
702 w/Tractor, VFU, List $2480 ...... $1995
703 w/Tractor, VFU, List $2975 ...... $2395
704 w/Tractor, VFU, List $2350 ...... $1995
TI 810 Basic, List $1895 .... $1695
810/serial & Centronics-style parallel interface, List $1940 .... $1735
810 w/full ASCII (U/LC), Vertical Forms Control, Compressed Print .. $1895
TI 820 KSR, List $2165 .... $1895
TI 745 w/full ASCII, List $1695 .... $1399
COMPRINT 912 w/parallel interface .. $ 559
912 w/serial interface, List $699 ...... $ 589
AXIOM IMP I .... $ 699
MICROTEK, List $750 .... $ 675
OKIDATA Microline 80, List $949 649
Tractor Feed Option .... $ 99
RS232 Serial Interface .... $ 89

NORTH STAR HORIZON®

HORIZON 1 ASSEMBLED & TESTED

32K, Double Density, List $2695 .... **$2279**
32K, Quad Density, List $2995 .... **$2539**

HORIZON 2 ASSEMBLED & TESTED

32K, Double Density, List $3095 ...... **$2619**
32K, Quad Density, List $3595 .... **$3049**
48K, Double Density, List $3590 ...... **$3039**
48K, Quad Density, List $4090 .... **$3469**
64K, Double Density, List $3830 ...... **$3239**
64K, Quad Density, List $4330 .... **$3669**

LIMITED QUANTITY OF HORIZON 2 KITS AVAILABLE

FLOPPY DISK SYSTEMS

NORTH STAR MDS-A
Assembled, List $899 ...... ***SPECIAL $ 719***
Kit Version, List $799 .... $ 669
MORROW THINKER TOYS® Discus 2D, List $1199 .... **OUR PRICE $1019***
Discus 2D, dual-drive, List $1994 ..... **$1694***
Discus 2+2, A&T, List $1549 .... **$1319***
Dual Discus 2+2, A&T, List $2748 ... **$2335***
*Now includes CP/M® 2.2
MICROMATION Megabox, DD w/ 8" drives, 1-megabyte, List $2295 ..... **$1949**
2-megabyte, List $3095 .... **$2629**
MICROPOLIS 1041 MacroFloppy® w/enclosure (no P.S.), List $695 ...... **$ 625**
1042 MacroFloppy w/case & AC P.S. . **$ 709**
1053 Dual MetaFloppy® , List $1895 ... **$1695**

VIDEO BOARDS

I/O Mapped

SD COMPUTER VDB-8024, kit, List $370 **$319 †**
Assembled, List $470 .... **$ 399 †**
XITEX SCT-100K, Kit .... **ONLY $154.95**
SCT-100A Assembled **$174.95**
SSM VB2 I/O, Kit, List $199 .... **Call**
Assembled & Tested, List $269 ..... **Call**

Memory Mapped

SSM VB1C, 16x64, Kit, List $179
Assembled & Tested, List $242
SSM VB3, 80-Char., 4MHz, Kit, List $485 **Call**
4 MHz, A&T, List $565 .... **Call**
INTERSYSTEMS, 16x64, A&T, List $165 **$149**

ESCON CONVERSION FOR IBM SELECTRIC

Complete w/microprocessor controller and power supply. Factory built. User installs solenoid assembly or it can be done at Escon factory at nominal cost.

Parallel (TRS-80, Sorcerer, etc.), $575 **$514**
RS232 Standard Serial, List $599 ...... **534**
IEEE-488 (for PET), List $660 .... **584**
TRS-80 Cable **25**

CALIFORNIA COMPUTER SYSTEMS

280 CPU BOARDS List $299 .... **$269**
DISK CONTROLLER 2422 List $399 .... **$359**
32 CASE STATIC List $710 .... **$599**
64K DYNAMIC BOARD List $699 .... **$589**

CPU BOARDS

(assembled unless noted)

NORTH STAR Z80A (ZPB-A/A), $299 **$254**
CROMEMCO 4 MHz (ZPU-W), List $395 **$335**
4 MHz (SCC-W), List $450 .... **$382**
INTERSYSTEMS (formerly Ithaca Audio)
new Series II Z-80, 4 MHz, List $395 ... **$349**
SSM CB1 8080 A&T, List $252 ....
CB1A Kit, List $183 ....
CB2 Z-80, A&T, List $344 ....
CB2 Kit, List $260 ....
DELTA Z-80, with I/O **$289**
SD SBC-100, List $350 .... **$298**
SBC-100 Kit, List $295 .... **$250**
SBC-200, List $400 .... **$332**
SBC-200 Kit, List $320 .... **$272**

MEMORY BOARDS

32K SD ExpandoRAM Kit ONLY $249†

ONLY $159 without RAM chips

NORTH STAR 16K Dynamic RAM Board, A&T (RAM-16-A/A), List $499 .... **$420**
16K Kit Version, List $449 ... **SPECIAL $299**
32K A&T (RAM-32/A), List $739 ...... **$620**
32K Kit, List $669 .... **SPECIAL $499**
CROMEMCO 16KZ-W, List $495 ...... **$419**
64KZ-W, List $1795 .... **1485**
MEASUREMENT SYSTEMS & CONTROLS
(Guaranteed performance, incl. labor/parts 1 yr)
DM6400 64K Board w/all 64K, $795 ... **$659**
DM4800 with 48K, List $695 .... **$589**
DM3200 with 32K, List $595 .... **$509**
DMB6400 64K Board w/all 64K **$859**
DMB4800 with 48K **$789**
MORROW SuperRAM — all static, all A&T
16K, 4 MHz or 2 MHz, List $349 .... **$299**
32K, 4 MHz, List $699 .... **$629**
16K Memory Master, List $399 .... **$339**
24K Memory Master, List $549 .... **$465**
INTERSYSTEMS (formerly Ithaca Audio)
8K Static 2 MHz, A&T, List $165 ...... **$149**
8K Static 4 MHz, A&T, List $195 ...... **$176**
16K Static 2 MHz, A&T, List $475 ..... **$427**
16K Static 4 MHz, A&T, List $495 ..... **$445**
64K Dynamic, List $995 .... **$895**
CALIFORNIA COMPUTER
16K Static, A&T, List $349.95 .... **$259**

FLOPPY DISK CONTROLLER BOARDS

NORTH STAR, DD, Assembled, List $499 .... **$399**
MORROW Disk Jockey 1, A&T ($213) . **$189**
Disk Jockey 2D, A&T, List $479 .... **$429**
SD Versafloppy 1, Kit, List $250 .... **$212 †**
Versafloppy II, DD Kit, List $350 ...... **$297 †**
Versafloppy II, DD, A&T, List $430 .... **$365 †**
DELTA double density A&T ($385) **$345**
CONDUCTOR, double density A&T ... **$269**
INTERSYSTEMS FDC-2, A&T, $495 ... **$439**
MICROMATION Doubler, DD, A&T ... **$399**
TARBELL Floppy Disk Interface Kit **$199**
double density, A&T, List $495 .... **$444**

NEW CROMEMCO DOUBLE DENSITY DISK CONTROLLER

List $595 ...... **OUR PRICE $505**

SHIPPING AND INSURANCE: Add $2.50 for boards, $6 for Selectric Converter or Floppy Disk Drives, $7.50 for Floppy Disk Systems, $15 for Horizon. SHIPPED FREIGHT COLLECT: SuperBrain, Centronics and T.I. printers. Contact us for shipping information on other terminals and printers.
Above prices reflect a 2% cash discount (order prepaid prior to shipment). Add 2% to prices for credit card orders, C.O.D.'s, etc. Prices are subject to change and offers subject to withdrawal without notice.

— WRITE FOR FREE CATALOG —

✓238

MiniMicroMart, Inc.

1618 James Street, Syracuse NY 13203 (315) 422-4467 TWX 710-541-0431

kilobaud Peterborough NH 03458

MICROCOMPUTING T.M. LIST OF ADVERTISERS

***This advertiser prefers to be contacted directly.**

kilobaud MICROCOMPUTING Peterborough NH 03458

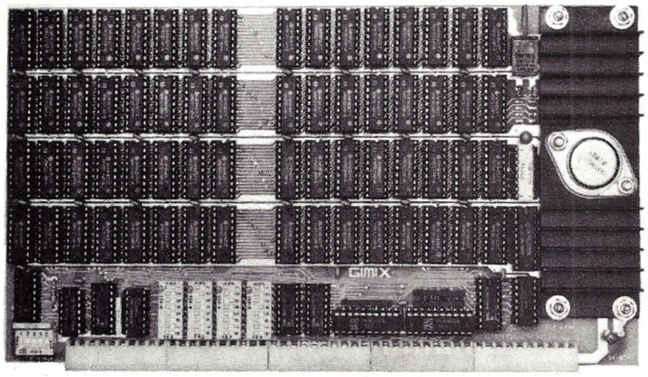
For further information from our advertisers, please use the Reader Service card located on the last page.

WHAT'S COOKING on the FIFTY BUS

32K STATIC RAM BOARDS

Designed for use with:

★ **Existing SS50 Systems** ★ **SS50C Extended Address Systems**



- Assembled
- Burned In
- Tested

16K . . . $328.12
24K . . . $438.14
32K . . . $548.15

16K and 24K Versions are socketed for 32K and require only additional 2114's for expansion.

FEATURES:

- Decoding for 4 Extended Address Lines (allows memory decoding up to 1 megabyte)
- DIP-switch to set extended addressing or disable it
- 4 separate 8K blocks, addressable to any 8K boundary by DIP-switch
- Each 8K block may be individually disabled
- Write protect either of two 16K sections
- Low power consumption — uses 2114L low power RAMS
- Fully Socketed
- Gold Bus Connectors
- Guaranteed 2MHz operation

AND NOW . . . GIMIX OFFERS YOU A Choice of 6800 or 6809 CPU CARDS

You can order your system to fit your needs or select one of the below featured systems. Please contact the factory for further information and availability.

Add as much memory as you need using **GIMIX Static RAM Cards** for the utmost in reliability.

32K 6800 SYSTEM $1,694.59
Includes: Chassis, 6800 CPU, 32K RAM BOARD, I/O card

32K 6809 SYSTEM $1,844.69
Includes: Chassis, 6809 CPU, 32K RAM BOARD, I/O card

32K 6809 PLUS SYSTEM $1,994.79
Includes: Chassis, 32K RAM BOARD, I/O Card, and features our 6809 PLUS CPU Card with the Time of Day Clock option with battery back-up installed, as well as the 6840 Timer Package that provides 3 independent 16 bit counters.

This system also allows the following options to be added at additional cost:

- Battery back-up of the 1K RAM by substituting CMOS parts.
- A 9511 or 9512 Arithmetic Processor.
- GIMIX or SWTP Dynamic Address Translators.

EXPORT NOTES:

For 50Hz 230V C.V. POWER SUPPLY . . . . Add $30.00
80 x 24 VIDEO BOARDS — Specify Format (No Added Charge)

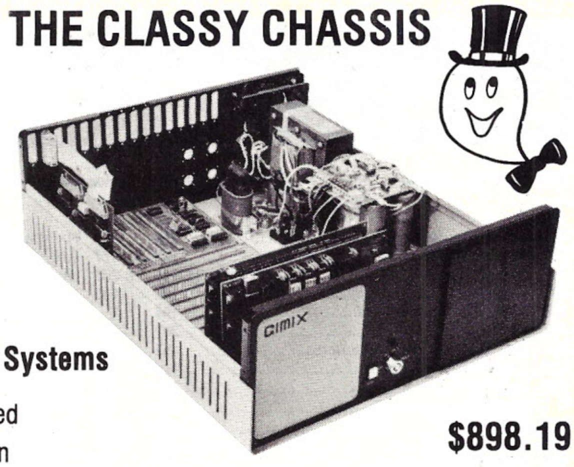
On Orders under $250.00 for a Single Board, or Chips, please Add $30.00 Handling and we will ship Air Mail Prepaid. On all other orders we will ship via Emery Air Freight Collect, and we will charge no handling. All orders must be prepaid in U.S. Funds. Please note that foreign checks have been taking about eight weeks for collection, so we would advise wiring money or checks drawn on a bank account in the U.S. Our bank is the Continental Illinois National Bank of Chicago, Account #73-32033. Visa or Master Charge also accepted.

FACTORY PRIME STATIC RAMS

2114L 450 ns . . $5.90 300 ns . . $6.40 200 ns . . $6.90
4044 450 ns . . $5.90 250 ns . . $6.90
ADD $5.00 HANDLING ON ORDERS UNDER $200.00

GIMIX® and **GHOST®** are Registered Trademarks of GIMIX INC.

THE CLASSY CHASSIS



$898.19

- 25 amp (5V) ferro-resonant constant voltage power supply,
- Heavy weight aluminum cabinet with 3 position key switch, fan, and provisions for two 5'' disk drives;
- 6800/6809 Mother Board, fifteen 50 pin and eight DIP-switch addressable 30 pin slots (gold plated pins), fully decoded;
- Baud rate generator on I/O section of Mother Board.

I/O BOARDS

for the 30 PIN BUS:

1 Port Serial . . . . (RS 232 or 20MA, current loop)	**$ 88.41**
2 Port RS 232 Serial	**128.43**
2 Port Parallel	**88.42**

for the 50 PIN BUS:

8 Port RS 232 Serial	**288.40**
8 Port RS 232 Serial . . . . with on board Baud Rate generator.	**318.46**
8 Port Parallel	**198.45**

BOTH 6809 SYSTEMS FEATURE OUR NEW TERMINAL BASED GMXBUG 09 SYSTEM MONITOR

GMXBUG 09 includes advanced debugging tools, utility, and memory manipulation routines.

Both 6809 Systems:

★ Can be reconfigured to allow use of other system monitors (OS-9 and SBUG-E)

★ Include 1K of Scratchpad RAM on the CPU

★ Allow optional software switching of system monitors.

2MHz 6809's at slight additional cost when they become available.

Phone, write, or see your dealer for details and prices on our broad range of Boards and Systems for the SS50/SS50C bus and our AC Power Control Products for all computers.

✓22

GIMIX INC.

The Company that delivers
Quality Electronic products since 1975.

1337 WEST 37th PLACE, CHICAGO, IL 60609
(312) 927-5510 • TWX 910-221-4055

Educator, Entertainer, Accountant.

Your Challenger Personal Computer.

Through the miracle of modern technology, a complete computer as powerful as the multimillion dollar room-sized computers of a few years ago can be put in a package the size of a typewriter and sells for as little as a color television set!

Through its years of microcomputer experience, Ohio Scientific has effectively channeled this tremendous computer power into a "friendly" computer with hundreds of personal uses, via a huge software library of programs for a broad range of personal, home, educational and business use.

This available software allows you to use and enjoy your computer without becoming an expert. The Challenger, however, is a powerful, general purpose computer which can be programmed in several languages by those who choose to.

Here are just a few of the popular uses of an Ohio Scientific Challenger Computer:

Education

The personal computer is the ultimate educational aid because it can entertain while it educates. Software available ranges from enhancing your children's basic math, reading and spelling ability, through tutoring high school and college subjects, to teaching the fundamentals of computers and computer programming.

Entertainment

Many of the Challenger's games educate while they entertain, from cartoons for preschoolers to games which sharpen mathematical and logical abilities. But, entertainment doesn't stop here. The Challenger's graphics capabilities and fast operation allow it to display action games with much more detail than the best video games, providing spectacular action in games such as Invaders, Space Wars, Tiger Tank and more! All popular sports such as golf, baseball and bowling are available as simulated computer games as well as many conventional games such as chess where the computer plays the role of a formidable opponent.

Accounting

Your Challenger computer can keep track of your checkbook, savings account, loans, expenses, monitor your calorie intake and your biorythms.

If you are involved in a business, you can use it to do word processing; accounting, inventory control, order processing, customer lists, client records, mailing labels and planning.

And more:

This may seem like a lot of uses, but it's only the tip of the iceberg for a general purpose computer. For example, your Challenger can be expanded to control lights and appliances, manage your energy usage and monitor for fire and break-ins. Furthermore, it can communicate with you, with other computers and the new personal computer information services over the telephone.

In fact, the uses of general purpose, personalized computers are expanding daily as more and more people discover the tremendous capabilities of these new technological wonders.

Ohio Scientific offers you four personalized computer systems starting at just $479.

For a free catalog and the name of the dealer nearest you, call 1-800-321-6850 toll free.

OHIO SCIENTIFIC

1333 SOUTH CHILLICOTHE ROAD
AURORA, OH 44202 • (216) 831-5600